Vivesvaraya Technological University
"Jnana Sangama", Belagavi - 590018

A Major Project Report on
(IS0602)

Travel Review Based Recommender System

*Submitted in partial fulfilment of the requirement
for the award of the degree of*

**BACHELOR OF ENGINEERING
IN
INFORMATION SCIENCE AND ENGINEERING**

Submitted by

Devi Prajwala B S	**4NI17IS023**
Harshita J	**4NI17IS027**
N S Sahana	**4NI17IS044**
Ambika M	**4NI18IS402**

Under the Guidance of
Mr Chandan J,
Assistant Professor,
Department of ISE, NIE,Mysore.

The National Institute of Engineering
(Autonomous Institute)
Mysore - 570008.

Department of Information Science and Engineering
NIE, Mysore - 570008.

2020-2021

The National Institute of Engineering
Manandavadi Road, Mysuru - 570008
Department of Information Science and Engineering

Certificate

Certified that the Major Project(IS0602) work entitled
"Travel Review Based Recommender System"
is the bona fide work carried out by

Devi Prajwala B S	**4NI17IS023**
Harshita J	**4NI17IS027**
N S Sahana	**4NI17IS044**
Ambika M	**4NI18IS402**

in partial fulfilment for the requirements of the eighth semester BE in Information Science & Engineering prescribed by The National Institute of Engineering, Autonomous Institution under Visvesvaraya Technological University, Belagavi. It is certified that all corrections and suggestions indicated by Guide have been incorporated. The Major Project report has been approved as it satisfies the academic requirements in respect of the project work prescribed for the eighth semester.

Signature of Guide	**Signature of HoD**	**Signature of Principal**
Mr.Chandan J	**Dr.P Devaki**	**Dr.N V Raghavendra**
Assistant professor	Head of the Department	Principal
Dept of ISE, NIE	**Dept of ISE, NIE**	**NIE, Mysore**

Name of the Examinar	**Signature with Date**
1.	
2.	

ABSTRACT

With advance in science and technology, finding solution to the problem of information storage and retrieval has become new area of expertise. Recommendation system is one of the widely accepted and best technique which can be effectively used to untangle these problems. Recommender system can help the users to take right decisions about the products they want to use and the services they want to enjoy.

A recommender system is mainly intended to provide personalized user experience. Travel review based recommender system , which is mainly based on statistical calculations and predictions can help to enhance user experience by finding the similarity of one user with other user. In this project, we are taking the user ratings for some of the places and using Pearson Similarity measure we are comparing that active user's ratings with the users who had already rated those places. We are making use of nearest neighbor concept to compare the new user's taste with that of other user's in database.

By using the inputs provided by the current user , the result will be displayed saying whether the remaining unvisited place can recommended to the current user or not . It will display whether the user likes or dislikes the place. This will reduce the time taken to retrieve the information and helps to take decisions in effective manner.

We are also integrating travel recommender system with recommending advertisements to the user. Naïve Bayes formula is applied to the inputs given by the user such as his age , income, internet usage and time spent online . We are making use of these values to determine whether the user is interested in the advertisements or not.

A recommender system is a growing technology which will reduce the amount of time consumed in information retrieval by only extracting the useful and helps to reply for the user specific queries. And user can experience a better quality of service that he may be interested in.

ACKNOWLEGEMENTS

The satisfaction that accompanies the successful completion of any task would be incomplete without the mention of people whose ceaseless cooperation made it possible whose constant guidance and encouragement crown all efforts.

We are extremely thankful to **Dr. N V Raghavendra** , Principal, NIE, Mysuru, for providing us the academic ambience and laboratory facilities to work, and everlasting motivation to carry out this work and shaping our careers.

We express our profound gratitude to **Dr. P Devaki**, HoD, Dept. of ISE, NIE, Mysuru, for her stimulating guidance, continuous encouragement and motivation throughout the course of present work.

We articulate our sincere regards to project coordinator, **Mrs. Prajakta M**, Assistant Professor, NIE, Mysuru, for her support and guidance over the entire course of work.

It gives us immense pleasure to thank our guide **Mr. Chandan J**, Assistant Professor, Department of ISE, NIE, Mysuru, for their valuable suggestions and guidance during the process of the project and for having permitted us to pursue work on the subject, special thanks for introducing the latex tool so that we could easily bulid the project report.

We would like to thank and express our gratitude to **Mr. Suhaas K P**, Assistant Professor,Department of ISE, NIE, Mysuru, for the continuous support and encouragement throughout the process of successful completion of the Major Project.

We take this opportunity to thank all our friends, classmates who always stood by us in difficult situations also helped us in some technical aspects and last but not the least, we wish to express deepest sense of gratitude to our parents who were a constant source of encouragement and stood by us as pillar of strength for completing this work and course successfully.

We are obliged to all the teaching and non-teaching staff, NIE, Mysuru for their assistance and advice throughout the workflow of the project.

Devi Prajwala B S	4NI17IS023
Harshita J	4NI17IS027
N S Sahana	4NI17IS044
Ambika M	4NI18IS402

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Project Purpose

With advance in technology and internet facility, information storage and information retrieval has become a complex problem. Recommendation system has become one of the most important feature in almost all aspects of life. It is taking a major role in untangling these complex problems. By helping people to take right decisions about the products they want to buy, service they want to make use of or the place where they want to visit.

Travel review based recommender system helps users to find the places which they may like to visit. This recommender system is mainly based on statistical computation and prediction where the similarity is calculated by the reviews given by large set other users. we are making use of user review based recommender system which suggest place to new user based on Pearson Similarity measure by comparing his ratings to the places with other users ratings to that places.

A recommender system aims to provide users with personalized online product or service recommendations to handle the increasing online information overload problem and improve customer relationship management.Travel review based recommender system is mainly based on statistical computation and prediction. We make use of user based recommender system to suggest the place to new user based on the Pearson similarity measure by comparing his ratings to the other places with other user's present in the data set. It helps us to make prediction whether the user would like or dislike the unseen place as the data we are using is the travel ratings.

1.2 Existing System

Recommender system is a kind of information filtering system which only gives the required and useful information to the user. This helps user to take right decision in a reduced amount of time. It captures users taste and recommends products or services that they may like. Thereby enhance user experience.

This system helps to filter information. By drawing from huge data sets obtained from either from the night crawler or the reviews taken as data being fed to the model thus prepared in order to obtain the next possible outputs that the user may be interested in can be therefore recommended.The system's algorithm can pinpoint accurate user preferences. Once you know what your users like, you can recommend them new, relevant content.

There are already many applications which successfully implemented recommender systems and giving best user experience. YouTube, an online video sharing and social media platform recommends videos to its user on the basis of their search history and previously watched videos. Spotify an audio streaming application which captures users emotion and their taste of music and suggest accordingly. Amazon, which is a e-commerce platform give recommendation based on user's purchased or rated items. It uses item based collaborative filtering. Netflix, online video streaming application which learn from its own users interest and time spent for watching and suggest further based on that.

A recommender system that can assist for various aspects about travelling such as destination , climate, transportation and photography for a specified destination. Advanced text analysis and opinion mining techniques enable the extraction of various types of review elements. Some examples of recommender systems in action include product recommendations on Amazon, Netflix suggestions for movies and TV shows in your feed, recommended videos on YouTube, music on Spotify, the Facebook newsfeed and Google Ads.

1.3 Proposed System

Travelling is a fusion of transportation, travel time, climate , journey, housing etc. To enjoy all these facilities, people look for travel assistance which take care of all these things. Lot of information is available online on travel assistance. So it is very difficult to search for each and every aspect and will take a lot of time to come up with a decision.

A travel review based recommender system, which analyze the users input using the technique of Pearson similarity Measure. It compare the the rating of current user with ratings given by large set of users in the database and extract the similarity. The steps will be as follows:

- To give ratings for the places which user already visited and the ID of the current (active) user as an input.
- Identify and compare the similarity with other user's review from the database who had similar preferences to those of the active user in the past.
- Then, for every place that the active user has not yet seen, a prediction is determined based on the ratings for places made by the peer users.

After the calculation, output will be shown whether the current user will like that unvisited place or not.

Chapter 2

Literature Survey

A recent advance in technology recommendation has become an essential. It is the process of automatically identifying the use of services and proactively discovering and recommending services to the end users. Effective recommendations should fulfill needs of both functional and non-functional requirements of the users[1]. The principal purpose of recommender system is to give personalized user recommendations by considering their taste of choosing products or services. Thereby enhancing the user experience. Search engines can find lots of information as required by user but processing that amount of information is very difficult task. Recommendation system tackles this issue by filtering and displaying only the specific information.

Collaborative Filtering is one of the most significantly used method for generating recommendations for users. Collaborative Filtering depends on selecting a subset of users called user neighborhood for filtering recommendations for current user[2]. Collaborative filtering prefers like minded users for giving recommendations to a new user. It finds the user in database who has given same ratings as current user and evaluate similarity between them by comparing the coefficient. A Pearson Correlation Coefficient formula can be used to predict the nearest neighbors of the user. It is a most widely used and accurate similarity measure used in collaborative filtering recommender systems[3].

Collaborative filtering has two types, memory-based and model-based. Memory-based Collaborative filtering gives excellent accuracy as compared model-based Collaborative filtering because in memory-based Collaborative filtering, the entire database is accessed every time a prediction is made. The memory-based Collaborative filtering computes the similarity with the active user by weighting all users in the neighborhood and selecting a subset of users having similar preferences as the active user.

There are two methods for neighborhood formation, the n nearest neighbor, where n is the number of nearest neighbors for a given user according to the similarity formula and the second method for neighborhood is threshold-based neighbor where the neighborhood is formed by calculating the distance between the active user and other users and its considered a neighbor when user similarity exceeds a given specified threshold[2].

Naive Bayes Classifier is used to calculate posterior probability. It is mostly used to process large volume of data. It is a very fast and efficient classifier. Bayes theorem with the independence assumptions between predictors. A Naive Bayesian model is easy to build, with no complicated iterative parameter estimation which makes it particularly useful for very large datasets. From this Naive Bayesian classifier, we have extracted the the formula for normal distribution which is applied on the continuous probability distribution for the real time random values of the dataset, this can be also termed as Gaussian. The Naive Bayesian classifier often does surprisingly well and is widely used because it often outperforms more sophisticated classification methods[6].

Chapter 3

System Requirement

3.1 Software Requirements

The code, equations, visualizations, narrative text written and compiled along with data transformation, numerical simulation, statistical modeling, data visualization, machine learning we have used visual studio code. The following gives the list of all the software tools along with languages used:

- The language used in our project is c++, we have included the header file bits/stdc++.h , is basically a header file which includes every other standard libraries needed for the code to execute successfully.Using this file reduces the time wasted in doing chores of including all the other files which we maybe interested in the due course. This also reduces all the major work of writing all the necessary header files, instead using this e can have all of them at once.We can even have all the STL of GNU C++ for every function we use.

- Another header file we have used is fstream which is helpful for the manipulation of the input and the output of the file management system. Where ifstream which is used for the stream class in order to read the inputs , where as ofstream is used for the stream class in order to write on to the files. Therefore using fstream includes both the read and write to the files. In our project it is necessary as the input dataset in redirected to the model to build it efficiently.

- In order to run the program we have used the shell script.This is the command line interpreter. Most important actions or functions performed by shell scripts which involves file manipulation, program execution, and printing the text. A script which sets up the environment, executes the program, and does any necessary cleanup, logging required, is called a wrapper. This script hence is used to get us the graphical user interface which is the front part of our project. Hence which allows the user to take his/her choices as to whether he/she will provide their own dataset or whether they are ready to consider the default dataset in order to train the model for the further prediction. the procedure is followed in the order like first the time spent by an individual on the internet and then which place maybe recommended to that person is checked. The graphical user interface for all these is provided once the shell script runs.In our project the script is run.sh

- As such including the shell script, we have also included the dialog package, as of for the use to display the contents to the user front end , to provide the graphical user interface like feature to the individual. This package, is available in the command line interpreter to ease the usage of the people and thus making it user friendly.It is much more lightweight than the official and more common graphical user interface that we use , but still providing us many of the intricacies and tricks which can be manipulated in GUI otherwise can now be possible using this dialog package.

- We have also incorporate Visual Studio, Linux subsystem, GNU debugger and GitHub, for the software tools.The visual studio code is used as the editor our c++ code. The Linux subsystem in order to run the script, adding to the GNU debugger to check for errors and what might possibly be causing these errors.

- Microsoft's Visual Studio Code is a source-code editor for Windows, Linux, and macOS. Debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git are among the features. Users can customise the theme, keyboard shortcuts, and preferences, as well as instal extensions that offer new features.

- The GNU Debugger (GDB) is a portable debugger that operates on a wide range of Unix-like systems and supports a wide range of programming languages, including Ada, C, C++, Objective-C, Free Pascal, Fortran, Go, and a few more. GDB has a lot of features for tracing and changing how computer programmes are executed. The user can monitor and change the values of internal variables in programmes, as well as call functions that are not part of the program's normal activity.

3.2 Hardware Requirements

The following are the Hardware Requirements for the running the system

- Hard Disk : 50 GB or more
- Ram : 4 GB and above
- Processor : i5 / AMD Fx 4100 (or greater)
- Processor Speed : 1.5 GHz or higher
- Operating System : Windows or Linux

Chapter 4

System Design

4.1 Introduction

The main aim of collaborative recommendation approach is to make use of the past behaviour or opinion of the users. The Pearson correlation coefficient takes values ranging from +1 to 1. Where +1 is considered as strong positive correlation and -1 is considered as strong negative correlation. Based on this calculated value the users who are similar to the current user is found by analyzing their past behaviour.

Pearson measure considers the fact that users are different with respect to how they interpret the rating scale. Some users may only give high ratings to the places they visit and some people may always give low ratings to the places. Pearson's correlation coefficient measure averages these ratings and makes it easier to compare the users review. That is through absolute values of ratings.

If we consider the whole project in hand and try to analyse it, it can deduced to few of the main steps that actually occur to support the the building of the model to produce effective results, therefore, we start with reading the dataset from a file where the required features are present separately for each case be it advertisement or place prediction. These data are thus passed on to the train the model , to get similarities of the expected output to that of the predicted one. These training helps the model to know about the possibilities and accuracy the model can perform with, the user input is then provided which sent to the stream of inputs and later fed to the built model, after the all data manipulations and statistical analysis, the output is provided which can be then displayed to the individual.

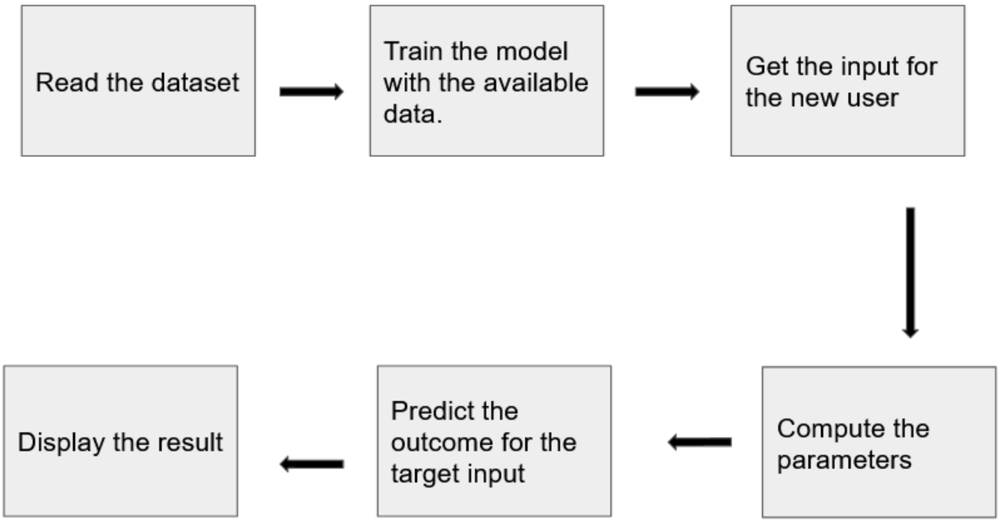


Figure 4.1: Flowchart of overall project

For evaluating prediction we have only considered those set of users in the database who has positive correlation with the current user. For reducing the size of neighbourhood we have included minimum threshold value for user similarity to limit the size for considering k nearest neighbours.

4.2 Working of Pearson's Model

In this method we make use of user based recommender system to suggest the place to new user based on the Pearson measure by comparing his ratings to the other places with other users present in the data set. It helps us to make prediction whether the user would like or dislike the unseen place as the data we are using is the travel ratings.The threshold value should be decided correctly. If the similarity threshold is too high, the neighbours set will be small right prediction cannot be made. if the similarity threshold is too low, the neighbours set will be very large that it becomes difficult to compare with the current user. If the number of neighbours are large in numbers, that will induce noise into the prediction. And if it is too low then the predictions will be negatively affected.

The formula used here is

$$r = \frac{\Sigma(X-\overline{X})(Y-\overline{Y})}{\sqrt{\Sigma(X-\overline{X})^2}\sqrt{(Y-\overline{Y})^2}}$$

Where, $\overline{X}$ = mean of X variable
$\overline{Y}$ = mean of Y variable

Figure 4.2: Formula for pearson's similarity measure

For the Pearson measure recommendation we have made use of a dataset from the uci machine learning repository which has the ratings of the users to different place.

Tarvel Review Ratings Data Set

Download: Data Folder, Data Set Description

Abstract: Google reviews on attractions from 24 categories across Europe are considered. Google user rating ranges from 1 to 5 a

Data Set Characteristics:	Multivariate, Text	**Number of Instances:**	5456	**Area:**	N/A
Attribute Characteristics:	Real	**Number of Attributes:**	25	**Date Donated**	2018-12-19
Associated Tasks:	Classification, Clustering	**Missing Values?**	N/A	**Number of Web Hits:**	63486

Source:

Shini Renjith, shinirenjith '@' gmail.com

Data Set Information:

This data set is populated by capturing user ratings from Google reviews. Reviews on attractions from 24 categories across Europe

Attribute Information:

Attribute 1 : Unique user id
Attribute 2 : Average ratings on churches
Attribute 3 : Average ratings on resorts
Attribute 4 : Average ratings on beaches
Attribute 5 : Average ratings on parks
Attribute 6 : Average ratings on theatres
Attribute 7 : Average ratings on museums
Attribute 8 : Average ratings on malls
Attribute 9 : Average ratings on zoo
Attribute 10 : Average ratings on restaurants
Attribute 11 : Average ratings on pubs/bars
Attribute 12 : Average ratings on local services
Attribute 13 : Average ratings on burger/pizza shops
Attribute 14 : Average ratings on hotels/other lodgings
Attribute 15 : Average ratings on juice bars

Figure 4.3: Dataset of travel review

Taking into account of the user based recommender systems, using the

Pearson's correlation, we initially take the dataset as comma separated values(csv) is put into the model,for training purpose, the mean for all the values is calculated as for the first step,further on we gather the new user's input from the respective individual of whom we want the recommendation of the tenth place for. Using the Pearson's correlation we are bound to calculate the similarity between the new user and those of the mean of the other users in the dataset. With these similarities under consideration we tend to conclude the results such that the new get recommended for the last place as to whether he will chose or not, we do this by keeping a threshold such that any value higher than that is considered and that value which is is the highest of all the above , we check it with our obtained similarity , the value which is the highest for the similarity is then considered. We then predict the output which will be then displayed on the screen using the dialog package to the user.

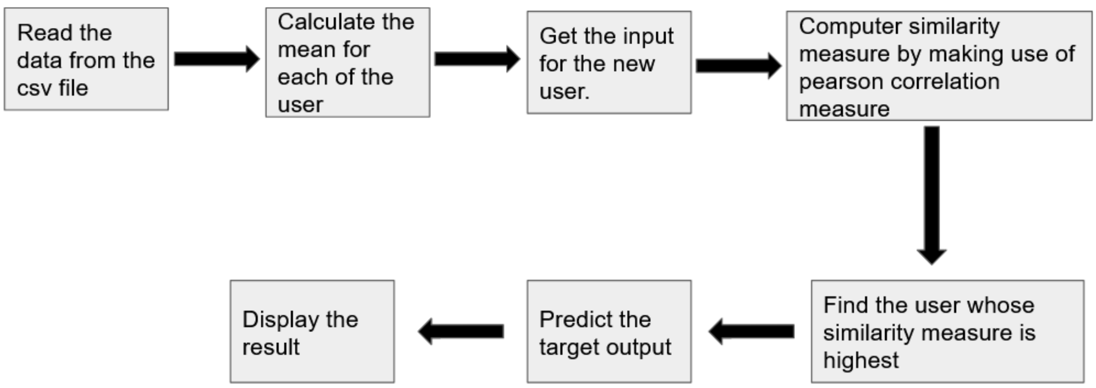
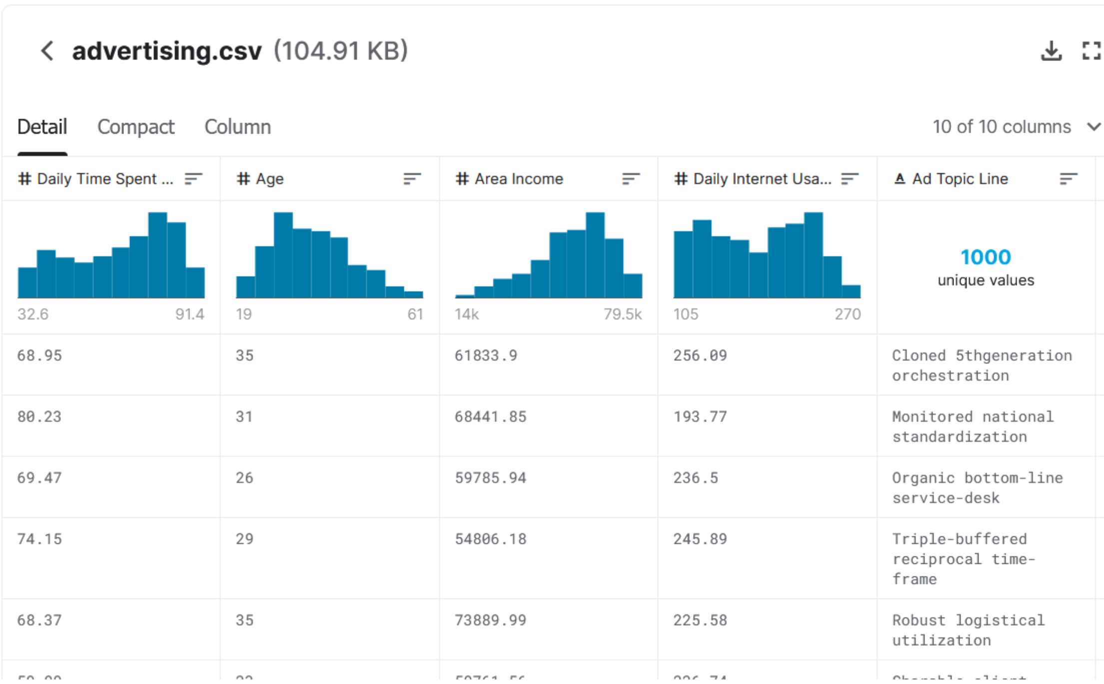


Figure 4.4: User based recommendation based on pearson's correlation coefficent flowchart

4.3 Working of Naive Bayes Classifier

The Naive Bayesian classifier is based on Bayes' theorem with the independence assumptions between predictors. A Naive Bayesian model is easy to build, with no complicated iterative parameter estimation which makes it particularly useful for very large datasets. Despite its simplicity, the Naive Bayesian classifier often does surprisingly well and is widely used because it often outperforms more sophisticated classification methods[6].For the naïve bayes algorithm we have used the advertisement dataset from the Kaggle which has attributes such as time spent in the web page,age,income,internet usage.

From this Naive Bayesian classifier, we have extracted the the formula for normal distribution which is applied on the continuous probability distribution for the real time random values of the dataset , this can be also termed as Gaussian.Using thesefunctions we calculate the gaussian probability density function since the variables are continuous , when the values of the attributes are given we can predict if he would click on the advertisement or not based on his internet usage, time spent, age, income. distribution.



# Daily Time Spent ...	# Age	# Area Income	# Daily Internet Usa...	Ad Topic Line
68.95	35	61833.9	256.09	Cloned 5thgeneration orchestration
80.23	31	68441.85	193.77	Monitored national standardization
69.47	26	59785.94	236.5	Organic bottom-line service-desk
74.15	29	54806.18	245.89	Triple-buffered reciprocal time-frame
68.37	35	73889.99	225.58	Robust logistical utilization

Figure 4.5: Dataset of advertisement data

We use Naive Bayes prediction for the recommender systems we take the dataset as comma separated values(csv) which were obtained form kaggle as mentioned before.

Is guided to the training to make sure that the model built is giving us the results that we wish to expect from a well trained statistical model, the we are supposed to split the dataset based on classes. Upon these data being fed to the model the mean and the standard deviation for each of the attributes of the each class, we read the input from the new user , in the order as given in the dataset, calculate the probability density function for the obtained value from the users, as per the probability density function the output is predicted based on the probabilities of the calculated values.

These gives us the insight as to whether the individual will click on the advertisement or is it more profitable or feasible for the advertisement to pop up to the user if the following conditions are met. the output is thus put forth for the new user in accordance with the values that they provided.

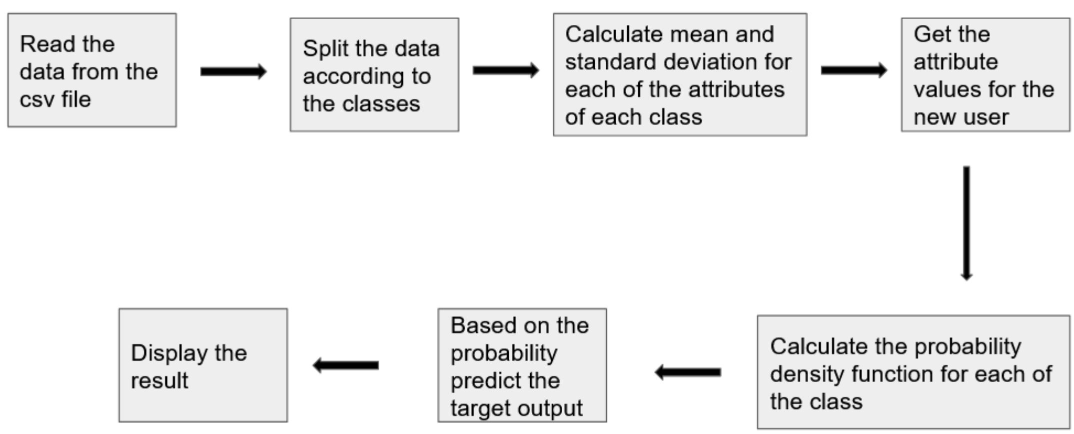


Figure 4.6: Naive bayes prediction to check whether the user clicks on the ad or not flowchart

In this method we calculate the Gaussian probability density function since the variables are continuous , when the values of the attributes are given we can predict if he would click on the advertisement or not based on his internet usage, time spent, age, income.

The formula used here is

$$\mu = \frac{1}{n}\sum_{i=1}^{n} x_i \qquad \boxed{\text{Mean}}$$

$$\sigma = \left[\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \mu)^2\right]^{0.5} \qquad \boxed{\text{Standard deviation}}$$

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \qquad \boxed{\text{Normal distribution}}$$

Figure 4.7: Naive Bayes prediction formula

Naive Bayes prediction, in this method we calculate the Gaussian probability density function since the variables are continuous , when the values of the attributes are given we can predict if they would click on the advertisement or not based on their internet usage, time spent, age, income.

Chapter 5

System Implementation

Pre-processing of the data before giving data as input to the model. Naive Bayes prediction, in this method we calculate the Gaussian probability density function since the variables are continuous , this in the case for predicting whether the new user will click on the advertisement or not given the desired features.

For evaluating prediction we have only considered those set of users in the database who has positive correlation with the current user, this in case of predicting whether the given person will be recommended with the tenth place or not based on the similarity measure.

5.1 Pearson's Similarity

Taking into account of the user based recommender systems, using the Pearson's correlation, we initially take the dataset as comma separated values(csv) is put into the model,for training purpose. We read the data fro the file as String Vector. we initialise the file pointer and point it to an existing file which is the imported data set or the data set provided by the user as per his choice.

This file is read from the input stream and then extracted row wise, where each row is read and stored to a string variable called 'line' which is then further broken into words. Every column is read of each instance which is then stored into another string variable called 'word', upon doing this we will be finally able to extract each of the column data of the row, which pushed to a vector named 'row'.Henceforth the data is processed in this format.

The function must be deployed in order to make the predictions necessary using the Pearson's similarity equations, where in which we use the mean values and the ratings of the user for the measurement. Each of the users average is taken and these values are subtracted with the average of the other users, these values is then summed up to get the value, upon putting the values into the Pearson's similarity equation to get the the measure, each of these similarity is then pushed to a vector called 'simili'.

The collaborative recommendation approach is to make use of the past behaviour or opinion of the users. The Pearson correlation coefficient takes values ranging from +1 to 1. Where +1 is considered as strong positive correlation and -1 is considered as strong negative correlation. Based on this calculated value the users who are similar to the current user is found by analyzing their past behaviour.

The main goal of the program is to get the predictions as to whether the new user will be recommended the place or whether the person will desire to go to that place or not, based on the similarity calculated earlier, we we tend to extract values greater than the given threshold which in our case we have taken a minimum of 2.5 , the user with the highest similarity will be considered , if the value is greater the given threshold then the place is recommended, or else it won't be recommended.

5.2 Naive Bayes Classifier

We use Naive Bayes prediction for the recommender systems we take the dataset as comma separated values(csv) which were obtained form kaggle as mentioned before. Is guided to the training to make sure that the model built is giving us the results that we wish to expect from a well trained statistical model, the we are supposed to split the dataset based on classes.

The function is written in order to split the data provided in the dataset in accordance with their class, since the number of attributes are 4 we try to run the in extent of its row, for each row we push a yes if the person will click on the advertisement or not, these are grouped these classes altogether.

For the calculation part we use both the mean and the standard deviation, and each of these values are stored in separate variables.For each column feature separately these values are calculated.

Naive Bayes prediction, in this method we calculate the Gaussian probability density function since the variables are continuous , when the values of the attributes are given we can predict if they would click on the advertisement or not based on their internet usage, time spent, age, income. Upon these data being fed to the model the mean and the standard deviation for each of the attributes of the each class, we read the input from the new user , in the order as given in the dataset.

Calculate the probability density function for the obtained value from the users, as per the probability density function the output is predicted based on the probabilities of the calculated values.These gives us the insight as to whether the individual will click on the advertisement or is it more profitable or feasible for the advertisement to pop up to the user if the following conditions are met. the output is thus put forth for the new user in accordance with the values that they provided.

5.3 Shell Script

We specify the script described in the run.sh shell script so as to run these algorithms simultaneously. These c++ codes for the respective algorithm are first compiled . The user is given with a chance to enter his own data or make use of the existing data. Then the naive Bayes algorithm is first compiled and later executed with output command embedded within the script to be run. If you have taken the choice that you yourself will provide the data, this will further evoke another code of c++ regarding the inclusion and considering the dataset manually provided by the new user to compile the algorithm as of now, but if the new user chooses not to do the above work then simply the existing dataset will be taken into consideration.

If the user has taken the chance that he will go with the default dataset provided by the program and not that he will provide the dataset himself, then the user has to input his own data to put to consideration, the function named ask data is called which will ask the user to give the values based on his view, which will be taken and further computed. The predicted data is thus finally displayed on the screen as to whether he will click the advertisement or not.

The user is thus asked whether he wants to be recommended or not, based on his answers, if yes he will follow the same procedure as earlier which is as follows ,the user is given with a chance to enter his own data or make use of the existing data. Then the naive Bayes algorithm is first compiled and later executed with output command embedded within the script to be run. If you have taken the choice that you yourself will provide the data, this will further evoke another code of c++ regarding the inclusion and considering the dataset manually provided by the new user to compile the algorithm as of now, but if the new user chooses not to do the above work then simply the existing dataset will be taken into consideration.

If the user has taken the chance that he will go with the default dataset provided by the program and not that he will provide the dataset himself, then the user has to input his own data to put to consideration, the function named ask data is called which will ask the user to give the values based on his ratings, which will be taken and further computed. The predicted data is thus finally displayed on the screen as to whether he will recommended the tenth place or not.

Chapter 6

Testing

Test Case no	Test Cases	Expected Result	Actual Result	Result state
1	Entering the values in the user interface window	Value is written to the text file	Value is written to the text file	Pass
2	Asking the user for the training data	User is allowed to enter	User is allowed to enter	Pass
3	Predicting whether user clicks on the advertisement	Prediction obtained	Prediction obtained	Pass
4	Ask ratings of 9 different places	User is allowed to enter the ratings	User is allowed to enter the ratings	Pass
5	Alphabet is entered for rating	Display warning and ask user to enter the rating again	Program crashes	Fail

Table 6.1: Tabulation of Test Cases 1

Test Case no	**Test Cases**	**Expected Result**	**Actual Result**	**Result state**
6	Allow user to enter daily time limit	Display window to enter the value	Display window to enter the value	Pass
7	Enter age greater than 200	Display error message	Program crashes	Fail
8	Display window and ask for recommendation of places	Display window with yes or no box	Display window with yes or no box	Pass
9	Ask user for the rating of first place	Allow user to enter the rating	Allow user to enter the rating	Pass
10	Enter symbol for rating in the window provided	Display error message	Program crashes	Fail

Table 6.2: Tabulation of Test Cases 2

Test Case no	**Test Cases**	**Expected Result**	**Actual Result**	**Result state**
11	Ask user for the rating of second place	Allow user to enter the rating	Allow user to enter the rating	Pass
12	Allow user to choose between yes or no button	Options can be chosen	Options can be chosen	Pass
13	Recommend places by considering the ratings	Display "can be recommended" message	Display "can be recommended" message	Pass
14	Enter alphanumeric character for rating in the window provided	Display error message	Program crashes	Fail
15	Ask user for the rating of third place	Allow user to enter the rating	Allow user to enter the rating	Pass

Table 6.3: Tabulation of Test Cases 3

Test Case no	**Test Cases**	**Expected Result**	**Actual Result**	**Result state**
16	Similarity calculation	Identify nearest neighbour	Identify nearest neighbour	Pass
17	Allow user to enter age	Display window to enter the value	Display window to enter the value	Pass
18	Ask user for the rating of fourth place	Allow user to enter the rating	Allow user to enter the rating	Pass
19	Enter the dataset by missing a value	Display error message	Program crashes	Fail
20	Recommend places by considering the ratings	Display ”cannot be recommended” message	Display ”cannot be recommended” message	Pass

Table 6.4: Tabulation of Test Cases 4

Test Case no	**Test Cases**	**Expected Result**	**Actual Result**	**Result state**
21	Ask user for the rating of fifth place	Allow user to enter the rating	Allow user to enter the rating	Pass
22	Display the predicted target outcome in the message box	Displays the message in the message box	Displays the message in the message box	Pass
23	Allow user to enter income	Display window to enter the value	Display window to enter the value	Pass
24	Enter daily time limit greater than 24hours	Display error message	Program crashes	Fail
25	Ask user for the rating of sixth place	Allow user to enter the rating	Allow user to enter the rating	Pass

Table 6.5: Tabulation of Test Cases 5

Test Case no	**Test Cases**	**Expected Result**	**Actual Result**	**Result state**
26	Ask user for the rating of eight place	Allow user to enter the rating	Allow user to enter the rating	Pass
27	Allow user to enter internet usage	Display window to enter the value	Display window to enter the value	Pass
28	Enter age in words	Display error message	Program crashes	Fail
29	Ask user for the rating of ninth place	Allow user to enter the rating	Allow user to enter the rating	Pass
30	Display the predicted target outcome of the recommendation in the message box	Displays the message in the message box	Displays the message in the message box	Pass

Table 6.6: Tabulation of Test Cases 6

Chapter 7

Deployment

Pre-processing of the data before giving data as input to the model. Naive Bayes prediction, in this method we calculate the Gaussian probability density function since the variables are continuous , this in the case for predicting whether the new user will click on the advertisement or not given the desired features.

For evaluating prediction we have only considered those set of users in the database who has positive correlation with the current user, this in case of predicting whether the given person will be recommended with the tenth place or not based on the similarity measure.

7.1 Pearson's Correlation Measure

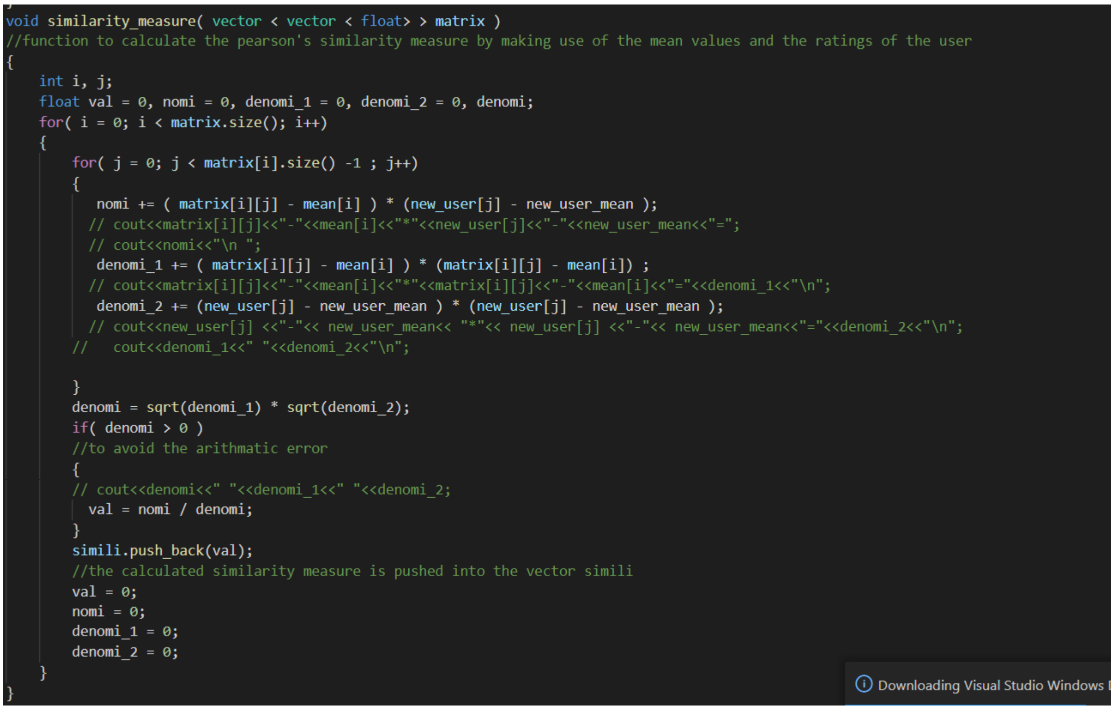
Taking into account of the user based recommender systems, using the Pearson's correlation, we initially take the dataset as comma separated values(csv) is put into the model,for training purpose. We read the data fro the file as String Vector. we initialise the file pointer and point it to an existing file which is the imported data set or the data set provided by the user as per his choice.

This file is read from the input stream and then extracted row wise, where each row is read and stored to a string variable called 'line' which is then further broken into words. Every column is read of each instance which is then stored into another string variable called 'word', upon doing this we will be finally able to extract each of the column data of the row, which pushed to a vector named 'row'.Henceforth the data is processed in this format.

```
void read_record( string data_name )
//read the Data from the file as String Vector
{

        // File pointer
        fstream fin;

        // Open an existing file
        fin.open (data_name, ios::in);

        string line, word, temp;

        while (fin >> temp)
    {
                // read an entire row and store it in a string variable 'line'
                getline(fin, line);

                // used for breaking words
                stringstream s(line);

                // read every column data of a row and store it in a string variable, 'word'
                while (getline(s, word, ',') )
        {
                        // add all the column data of a row to a vector
                        row.push_back(word);
                }
    }
}
```

Figure 7.1: Function to read data from csv file

The function must be deployed in order to make the predictions necessary using the Pearson's similarity equations, where in which we use the mean values and the ratings of the user for the measurement. Each of the users average is taken and these values are subtracted with the average of the other users, these values is then summed up to get the value, upon putting the values into the Pearson's similarity equation to get the the measure, each of these similarity is then pushed to a vector called 'simili'.



```
void similarity_measure( vector < vector < float> > matrix )
//function to calculate the pearson's similarity measure by making use of the mean values and the ratings of the user
{
    int i, j;
    float val = 0, nomi = 0, denomi_1 = 0, denomi_2 = 0, denomi;
    for( i = 0; i < matrix.size(); i++)
    {
        for( j = 0; j < matrix[i].size() -1 ; j++)
        {
          nomi += ( matrix[i][j] - mean[i] ) * (new_user[j] - new_user_mean );
         // cout<<matrix[i][j]<<"-"<<mean[i]<<"*"<<new_user[j]<<"-"<<new_user_mean<<"=";
         // cout<<nomi<<"\n ";
          denomi_1 += ( matrix[i][j] - mean[i] ) * (matrix[i][j] - mean[i]) ;
         // cout<<matrix[i][j]<<"-"<<mean[i]<<"*"<<matrix[i][j]<<"-"<<mean[i]<<"="<<denomi_1<<"\n";
          denomi_2 += (new_user[j] - new_user_mean ) * (new_user[j] - new_user_mean );
         // cout<<new_user[j] <<"-"<< new_user_mean<< "*"<< new_user[j] <<"-"<< new_user_mean<<"="<<denomi_2<<"\n";
        //   cout<<denomi_1<<" "<<denomi_2<<"\n";

        }
        denomi = sqrt(denomi_1) * sqrt(denomi_2);
        if( denomi > 0 )
        //to avoid the arithmatic error
        {
        // cout<<denomi<<" "<<denomi_1<<" "<<denomi_2;
          val = nomi / denomi;
        }
        simili.push_back(val);
        //the calculated similarity measure is pushed into the vector simili
        val = 0;
        nomi = 0;
        denomi_1 = 0;
        denomi_2 = 0;
    }
}
```


Figure 7.2: Similarity measure of pearson

The collaborative recommendation approach is to make use of the past behaviour or opinion of the users. The Pearson correlation coefficient takes values ranging from +1 to 1. Where +1 is considered as strong positive correlation and -1 is considered as strong negative correlation. Based on this calculated value the users who are similar to the current user is found by analyzing their past behaviour.

The main goal of the program is to get the predictions as to whether the new user will be recommended the place or whether the person will desire to go to that place or not, based on the similarity calculated earlier, we we tend to extract values greater than the given threshold which in our case we have taken a minimum of 2.5 , the user with the highest similarity will be considered , if the value is greater the given threshold then the place is recommended, or else it won't be recommended.

```
void make_prediction( vector < vector <float> > matrix )
//function to make the prediction whether the user would like or dislike the 10th place based on the similarity measure
{
    int max = 0, index = 0;
    for(int i = 0; i < simili.size(); i++)
    {
      if(max <= simili[i])
      //searching for the user who most similar to the new user i.e maximum pearsons co effecient
      {
          max = simili[i];
          index = i;
      }
    }
    if( matrix[index][9] >= 2.5 )
    //if the rating of the 10th place is greater than or equal to 2.5 then the 10th place is recommended else it is not recommended
    {
        cout<<"dialog --msgbox \" Place 10 can be recommended to the user\" 10 30\n";
       // cout<<"Place 10 can be recommended to the user\n";
    }
    else
    {
        cout<<"dialog --msgbox \"User may not like the 10th place\" 10 30\n";
        //cout<<"User may not like the 10th place";
    }
}
```

Figure 7.3: Function for prediction using Pearson Similarity Measure

7.2 Naive Bayes Classifier

We use Naive Bayes prediction for the recommender systems we take the dataset as comma separated values(csv) which were obtained form kaggle as mentioned before. Is guided to the training to make sure that the model built is giving us the results that we wish to expect from a well trained statistical model, the we are supposed to split the dataset based on classes.

The function is written in order to split the data provided in the dataset in accordance with their class, since the number of attributes are 4 we try to run the in extent of its row, for each row we push a yes if the person will click on the advertisement or not, these are grouped these classes altogether.

```
void splitting_classes()
//function to split the data in accordance with thier classes
{
    for(int i = 4 ;i < row.size() ;i += 5)
    //4 since the number of attributes is 4
   {
       int j, ind = i;
       if( stoi (row[i]) == 1 )
       //if the class is yes push the values into the vector row_yes
       {
           for(j = 4; j >= 1; j--)
           {
             row_yes.push_back( stof (row[ind-j]) );
             //push all the four values of attribute by converting them into float
           }
       }
       else if( stoi (row[i]) == 0 )
       //if the class is no push the values into the vector row_no
       {
           for(j = 4; j >= 1; j--)
           {
             row_no.push_back( stof (row[ind-j]) );
             //push all the four values of attribute by converting them into float
           }
       }
   }
}
```

Figure 7.4: Splitting classes in naive algorithm

For the calculation part we use both the mean and the standard deviation, and each of these values are stored in separate variables.For each column feature separately these values are calculated.

Naive Bayes prediction, in this method we calculate the Gaussian probability density function since the variables are continuous , when the values of the attributes are given we can predict if they would click on the advertisement or not based on their internet usage, time spent, age, income. Upon these data being fed to the model the mean and the standard deviation for each of the attributes of the each class, we read the input from the new user , in the order as given in the dataset.

Calculate the probability density function for the obtained value from the users, as per the probability density function the output is predicted based on the probabilities of the calculated values.These gives us the insight as to whether the individual will click on the advertisement or is it more profitable or feasible for

```
void calculation(vector <float> rows, float &time_spent_mean, float &age_mean, float &income_mean, float &internet_usage_mean,
                 float &time_spent_sd, float &age_sd, float &income_sd, float &internet_usage_sd )
{    for(int i=0;i<rows.size();i++)
        //calculate the sum of each of the attributes
    {
        time_spent_mean += rows[i];
        age_mean += rows[i+1];
        income_mean += rows[i+2];
        internet_usage_mean += rows[i+3];
        i+=3;
    }
    //clculate the mean for each of the attributes
        //mean = sum of the attribute values divided by the total number of attributes
        int n = rows.size ()/4;
    time_spent_mean /= n;
    age_mean /= n;
    income_mean /= n;
    internet_usage_mean /= n;
        for(int i = 0; i < rows.size(); i++)
        //calculate the variance for each of the attribute
        {
      time_spent_sd += ( rows[i] - time_spent_mean)*( rows[i] - time_spent_mean);
          age_sd += ( rows[i+1] - age_mean)*( rows[i+1] - age_mean);
          income_sd += ( rows[i+2] - income_mean)*( rows[i+2] - income_mean);
          internet_usage_sd += ( rows[i+3] - internet_usage_mean)*( rows[i+3] - internet_usage_mean);
          i+=3;
        }
    //calculate the standard deviation for each of the attribute
        //the stadard deviation is the square root of the variance
        time_spent_sd = sqrt (time_spent_sd / n );
        age_sd = sqrt (age_sd / n );
        income_sd = sqrt (income_sd / n );
        internet_usage_sd = sqrt (internet_usage_sd / n );
}
```

Figure 7.5: Computations in naive algorithm

the advertisement to pop up to the user if the following conditions are met. the output is thus put forth for the new user in accordance with the values that they provided.

```
float make_prediction(float time, float age, float income, float internet_usage, float time_spent_mean,float age_mean, float income_mean,
                      float internet_usage_mean, float time_spent_sd, float age_sd,  float income_sd, float internet_usage_sd)
{
        //the pdf(probability density function ) is calculated using the gaussian probability for the continous variables which helps in making the pediction whether the
        //user would click o the add or not

        float val, res;
        val = ( ( time - time_spent_mean ) * ( time - time_spent_mean ) ) / (2 * (time_spent_sd * time_spent_sd) );
        time_spent_pdf = (1 / (sqrt (2 * 3.1428) * time_spent_sd) ) * exp( -1*val);


        val = ( ( age - age_mean ) * ( age - age_mean ) ) / (2 * (age_sd * age_sd) );
        age_pdf = (1 / (sqrt (2 * 3.1428) * age_sd )) * exp( -1*val);


        val = ( ( income - income_mean ) * ( income - income_mean ) ) / (2 * (income_sd * income_sd) );
        income_pdf = (1 / (sqrt (2 * 3.1428) * income_sd)) * exp( -1*val);


        val = ( ( internet_usage - internet_usage_mean ) * ( internet_usage - internet_usage_mean ) ) / (2 * (internet_usage_sd * internet_usage_sd) );
        internet_usage_pdf = (1 / (sqrt (2 * 3.1428)) * internet_usage_sd ) * exp( -1*val);

    res = time_spent_pdf * age_pdf * income_pdf * internet_usage_pdf;

        return res;

}
```

Figure 7.6: Prediction in naive algorithm

7.3 Working of Shell Script

We specify the script described in the run.sh shell script so as to run these algorithms simultaneously. These c++ codes for the respective algorithm are first compiled . The user is given with a chance to enter his own data or make use of the existing data. Then the naive Bayes algorithm is first compiled and later executed with output command embedded within the script to be run. If you have taken the choice that you yourself will provide the data, this will further evoke another code of c++ regarding the inclusion and considering the dataset manually provided by the new user to compile the algorithm as of now, but if the new user chooses not to do the above work then simply the existing dataset will be taken into consideration.

```
#script to run the algorithm by taking various input by making use of windows
#echo -e "IMPLEMENTATION OF NAIVE BAYES ALGORITHM\n\n"

function ask_data
#function which asks the user to enter the input into the window
{
    dialog --backtitle "Dialog Form" --title "user input" \
    --form "\nEnter the details" 25 60 16 \
   "Daily time limit:" 1 1 "Value 1" 1 25 25 30 \
   "Age:" 2 1 "Value 2" 2 25 25 30 \
   "Income:" 3 1 "Value 3" 3 25 25 30 \
   "Internet usage:" 4 1 "Value 4" 4 25 25 30 \
2>user_input.txt
}


dialog --yesno "Do you want to enter your own data?" 10 20
#user is given the chance to enter his own data or make use of the existing data
if [ $? -eq 0 ]
then
    g++ naive_data.cpp -o naive
    #naive.cpp is compiled

    ./naive
    #program is executed

    #echo "The entered data is"
    echo "data.csv" > user_input.txt
    #string is written to text file

    g++ naive_algo.cpp -o algo
    #program is compiled

else
    echo "The existing data will be used"
```

Figure 7.7: Script for running and and compiling the program-1

If the user has taken the chance that he will go with the default dataset provided by the program and not that he will provide the dataset himself, then the user has to input his own data to put to consideration, the function named ask data is called which will ask the user to give the values based on his view, which will be taken and further computed. The predicted data is thus finally displayed on the screen as to whether he will click the advertisement or not.

```
    echo "advertising.csv" > user_input.txt
    #string is written to text file

    g++ naive_algo.cpp -o algo
    #program is compiled


fi
ask_data;
#function is called to enter data

./algo < user_input.txt > output.sh
#program is executed

./output.sh
#script is executed

#the above lines of code deal with 2 situations one when the user wants to enter his own dat


dialog --yesno "Do you want to check for the recommendation of places?" 10 20
#user is asked if wants the recommendation or not
if [ $? -eq 0 ]
then
    #script to run the pearson algorithm
    # echo -e "Pearson's prediction\n\n"


    function ask_data_pearson {
    #function which asks the user to enter the input into the window

    dialog --backtitle "Dialog Form" --title "user input" \
    --form "\nEnter the ratings" 25 60 16 \
   "Church:" 2 1 "Value 2" 2 25 25 30 \
   "Resort:" 3 1 "Value 3" 3 25 25 30 \
   "Beach:" 4 1 "Value 4" 4 25 25 30 \
```

Figure 7.8: Script for running and and compiling the program-2

The user is thus asked whether he wants to be recommended or not, based on his answers, if yes he will follow the same procedure as earlier which is as follows ,the user is given with a chance to enter his own data or make use of the existing data. Then the naive Bayes algorithm is first compiled and later executed with output command embedded within the script to be run. If you have taken the choice that you yourself will provide the data, this will further evoke another code of c++ regarding the inclusion and considering the dataset manually provided by the new user to compile the algorithm as of now, but if the new user chooses not to do the above work then simply the existing dataset will be taken into consideration.

```
"Beach:" 4 1 "Value 4" 4 25 25 30 \
"Park:" 5 1 "Value 5" 5 25 25 30 \
"Theatre:" 6 1 "Value 6" 6 25 25 30 \
"Meuseum:" 7 1 "Value 7" 7 25 25 30 \
"Mall:" 8 1 "Value 8" 8 25 25 30 \
"Zoo:" 9 1 "Value 9" 9 25 25 30 \
2>pearson_input.txt
}


dialog --yesno "Do you want to enter your own data?" 10 20
#user is given the chance to enter his own data or make use of the existing data
if [ $? -eq 0 ]
then
  g++ pearson_data.cpp -o pearson_data
  #program is compiled

  ./pearson_data
  #program is executed
 echo $n > pearson_input.txt
 echo "10\n" > pearson_input.txt

  g++ pearson_algo.cpp -o pearson
  #program is compiled

else
 echo "The existing data will be used"

 echo "google_review_ratings.csv\n" > pearson_input.txt
 echo "5456\n" >> pearson_input.txt
 echo "10\n" > pearson_input.txt
 #string is written to text file
```

Figure 7.9: Script for running and and compiling the program-3

If the user has taken the chance that he will go with the default dataset provided by the program and not that he will provide the dataset himself, then the user has to input his own data to put to consideration, the function named ask data is called which will ask the user to give the values based on his ratings, which will be taken and further computed. The predicted data is thus finally displayed on the screen as to whether he will recommended the tenth place or not.

```
    else
     echo "The existing data will be used"

     echo "google_review_ratings.csv\n" > pearson_input.txt
     echo "5456\n" >> pearson_input.txt
     echo "10\n" > pearson_input.txt
     #string is written to text file

     g++ pearson_algo.cpp -o pearson
     #program is compiled

     fi
     ask_data_pearson;
     #function is called to enter data

     ./pearson < pearson_input.txt > output.sh
     #program is executed

     ./output.sh
     #script is executed
fi
clear
```

Figure 7.10: Script for running and and compiling the program-4

Chapter 8

Results

We are also integrating travel recommender system with recommending advertisements to the user. Naïve Bayes formula is applied to the inputs given by the user such as his age , income, internet usage and time spent online . We are making use of these values to determine whether the user is interested in the advertisements or not.

A recommender system is a growing technology which will reduce the amount of time consumed in information retrieval by only extracting the useful and helps to reply for the user specific queries. And user can experience a better quality of service that he may be interested in.

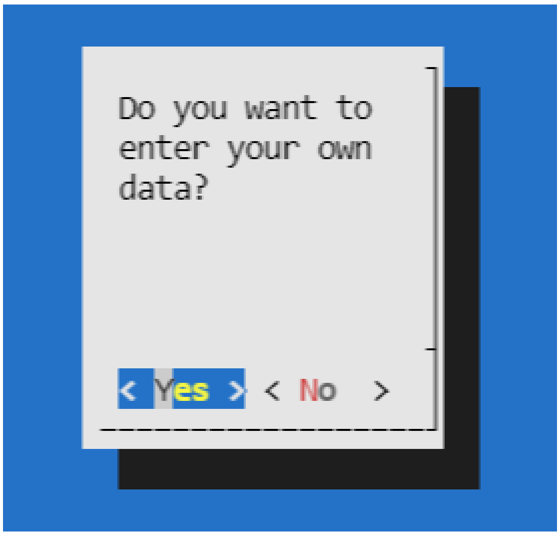


Figure 8.1: We want to use our own dataset or not

At first, a dialogue box will appear asking the user whether he wants to enter his own data for dataset or to use already existing dataset.

If user wants to enter his own data for the dataset, first he needs to specify how may entries he wants to insert and has to enter all other details.

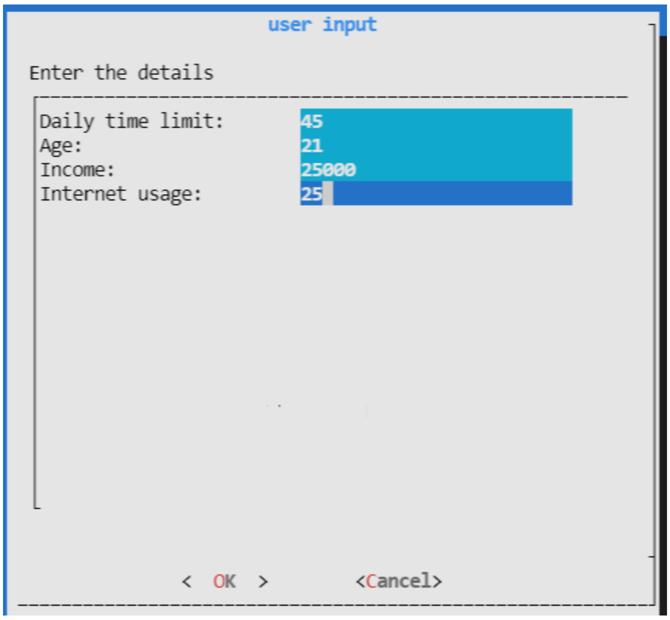


Figure 8.2: Enter the data for prediction

At first, we are taking the inputs from the new user such as daily time limit, his age, income and internet usage. These values are compared with other users in the dataset.

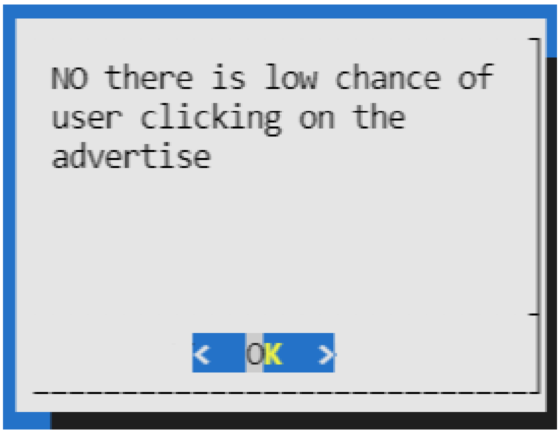


Figure 8.3: Low chance of user clicking on the add

After entering all the required information a dialogue box of result will appear displaying , whether there is a probability of user clicking on the advertisements. Above is the case where there is a Low probability of user clicking the advertisements.

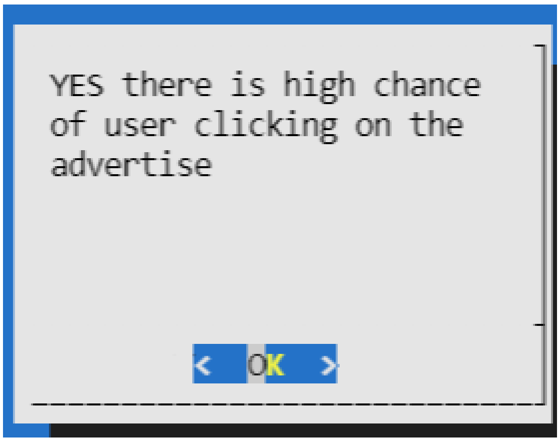


Figure 8.4: High chance of user clicking on the add

Above is the case where the probability of user clicking the advertisements is high.

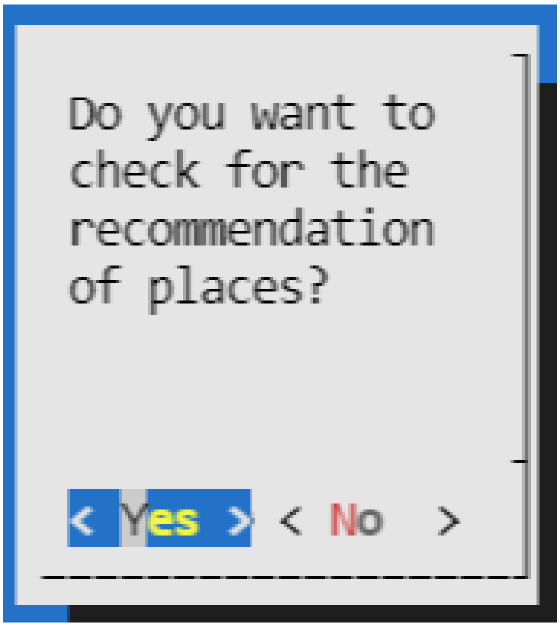


Figure 8.5: Ask for recommendation

After the output displayed for the advertisement, a window will be displayed for entering the ratings for some of the places if the user wants to continue for the travel recommender system.

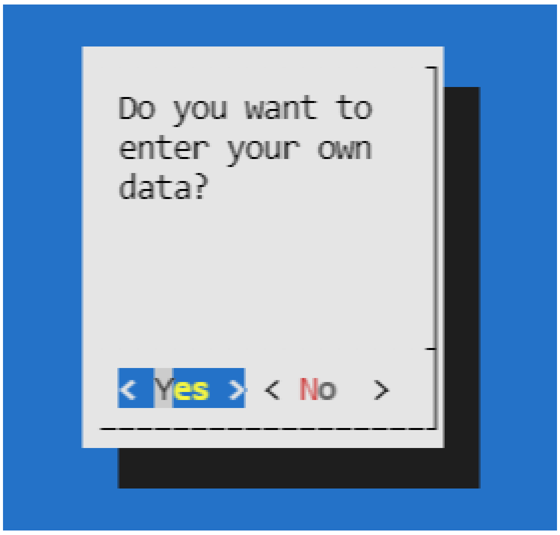


Figure 8.6: We want to use our own dataset or not

A dialogue box will appear asking the user whether he wants to enter his own data for dataset or to use already existing dataset. If user wants to enter his own data for the dataset, first he needs to specify how may entries he wants to insert and has to enter all other details. Otherwise existing dataset will be used.

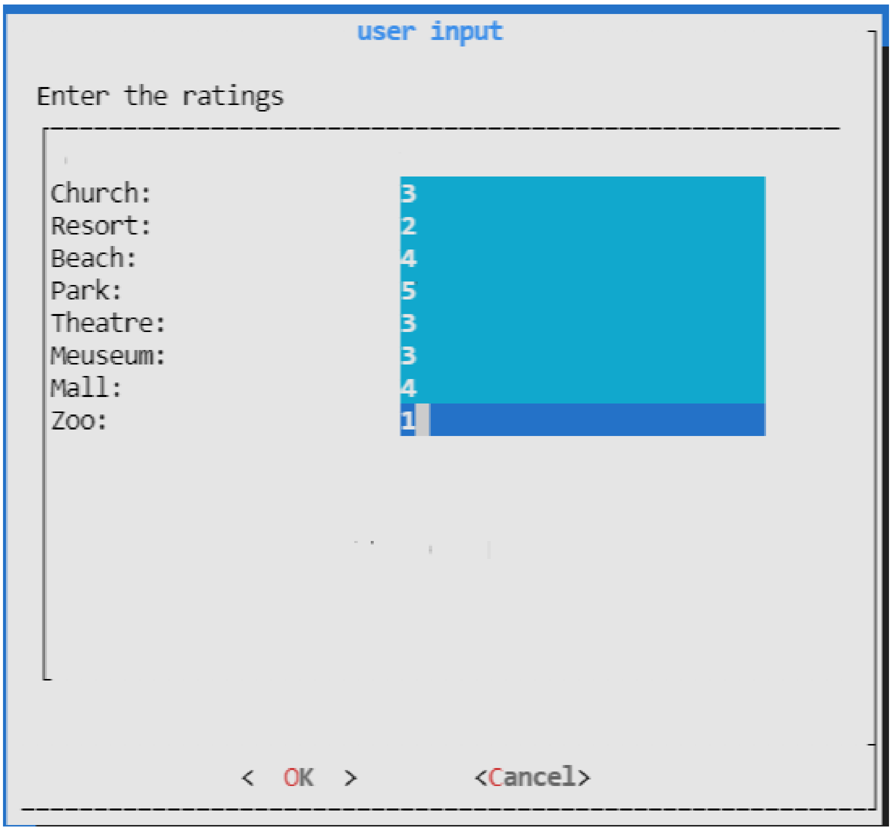


Figure 8.7: Enter the data for prediction

After continuing to travel recommender system, user needs to give ratings for all the places and needs to submit that.

After submitting the ratings for all the places, a result window will be displayed saying whether the user will like that remaining 10th place or not. Below output is when there is a possibility of user disliking the unvisited 10th place.

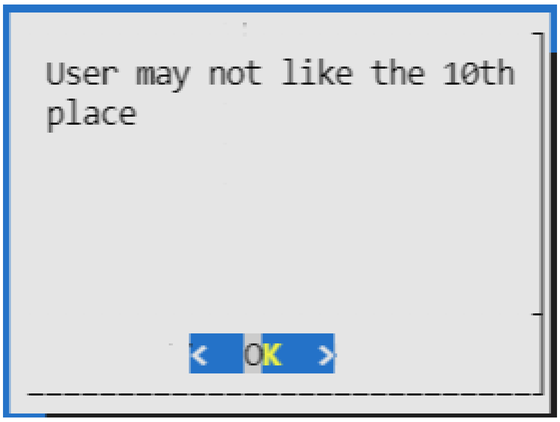


Figure 8.8: Low chance of user liking the place

Below is the case where the probability of user liking the place is high. And the unvisited place can be recommended to the user.

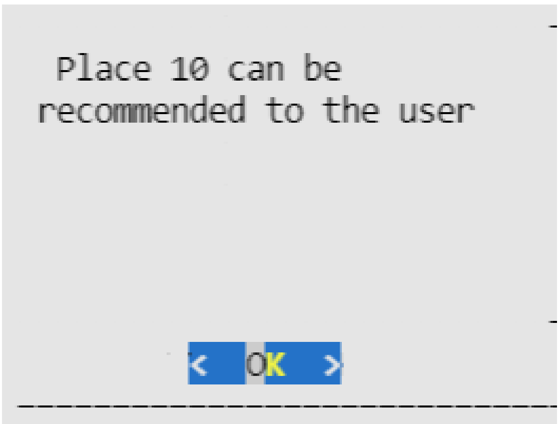


Figure 8.9: High chance of user liking the place

Chapter 9

Advantages and Disadvantages

9.1 Advantages

This system is useful for those people who visit new places quite often. People can easily decide whether the place is good or bad by using this application.Using "algorithm name" user will get recommendations of places. Since system ranks the feedback based on the weightage of the keywords in database, so the result is appropriate. User can decide which place to accommodate before they reach the place

The test statistic Pearson's correlation coefficient assesses the statistical link, or association, between two continuous variables. Because it is based on the method of covariance, it is known as the best method for quantifying the relationship between variables of interest. It reveals the amount of the association, or correlation, as well as the direction of the relationship.

This method not only establishes the presence or lack of correlation between any two variables, but also the exact magnitude or degree of connection. We may also determine the direction of the correlation using this method, that is, whether the correlation between the two variables is positive or negative.

This method uses regression equations to estimate the value of a dependent variable in relation to a specific value of an independent variable.

This approach contains a number of algebraic qualities that make calculating the co-efficient of correlation, as well as a number of other related quantities such as the coefficient of determination, simple.

Naive Bayes is simple and straightforward to apply.It does not necessitate as much data for training. Training does not demand as much data. It can handle a large number of predictors and data points. It is quick and can be used to make predictions in real time. It is unaffected by non-essential characteristics.

For multi-class prediction issues, Naive Bayes is a good choice. This algorithm is fast and can help you save a lot of time. If the premise of feature independence remains true, it can outperform other models while using far less training data. Categorical input variables are more suited to Naive Bayes than numerical input variables.

User based Collaberative filtering needs no requirement for domain knowledge. Because the embeddings are automatically taught, we don't need domain expertise. The model has the potential to assist users in discovering new hobbies. Even if the ML system does not know the user is interested in a certain item, it may nonetheless propose it since other users are interested in it.

To some extent, the feedback matrix is all that is required to train a matrix factorization model. The system, in particular, does not require contextual characteristics. This can be used as one of several candidate generators in practise. Even if no information about an item is available, we can nevertheless forecast its rating without having to wait for a user to buy it. Focusing entirely on material eliminates any flexibility in terms of the user's viewpoint and preferences. Captures small traits that are already present. This is especially true in the case of latent factor models. If the majority of users purchase two unrelated items.

9.2 Disadvantages

System will match the opinion with those keywords which are in database rest of the words are ignored by the system and It may provide inaccurate results if data entered incorrectly.

It is comparatively difficult to calculate since it necessitates the use of complex algebraic methods.It is highly likely to be misconstrued, especially if the data is homogeneous.

It is dependent on a number of assumptions, such as linear relationships, cause-and-effect relationships, and so on, which may or may not be true.

It takes a long time to get the results as compared to other approaches.The values of the extreme things have a big impact on it. It is subject to probable error, which its proponent recognises, hence computing probable error while interpreting its results is always recommended.

In Naive Bayes, all predictors (or traits) are assumed to be independent, which is rarely the case in real life. This limits the algorithm's usability in real-world scenarios. You shouldn't take its probability outputs seriously because its estimations can be off in some instances.

The 'zero-frequency problem' occurs when an algorithm assigns zero probability to a categorical variable whose category in the test data set was not present in the training dataset. To get over this problem, you should employ a smoothing approach.

The dot product of the associated embeddings represents the model's forecast for a given (user, item) pair. As a result, if an item isn't encountered during training, the system won't be able to construct an embedding for it or query the model with it. The cold-start difficulty is a term used to describe this issue.

If in case of the sparse matrix or not all user have provided ratings for all the places that are given in the dataset then this could lead to biased results or inaccurate results.

Chapter 10

Cost Effectiveness

Recommender system mainly deals with the likes and dislikes of the users, its major objective is to recommend an item to a user which has high chance of liking or in need for a particular user based on his previous purchases. It is like to have personalized team who could understand our likes and dislikes and help us in making the decisions regarding a particular item without biased by any means by making use of the large amount of data in the repositories which are generated day by day. The aim of recommender systems is to supply simply accessible, high-quality recommendations for the user community.It's wish is to own a reasonable personal authority with efficiency.

Earlier ,it can be said that the people would go to places based on some suggestions from their peers,but these days these can be achieved mush more easily from the reviews and the ratings of each of these places and based on this the recommendations can be done to the clients who have similar tastes. Hence this is much more time efficient and needs no much manual labor from the user perspective.

It is moreover like to have an personalised agent who could suggest us the items or things or movies or web series in our project it is the place based on analyzing our pattern of liking the items.

Decision making process is improved through travel recommender system. Time taken to search or explore for new places is reduced since user gets recommendation for the places that he may like. Some recommender systems also gives details on transportation which helps users to plan accordingly and economically.

Conclusion

A recommender system is a technique to filter the large about of information available. Only the information specific to user query is filtered out. Users interest is captured and that acts a filtering parameter for a recommender system. This helps users to take right decisions on what products or services they want to use and enjoy. User based collaborative filtering is a most widely used method for recommendation systems. Nearest neighbours selected from user based recommender system will be mostly similar as the taste of current user.

The travel review based recommender system also works in the same way to track current user interest by comparing their ratings given to the places that they had visited, with the available database of large set of other user's review to the same places. The users whose similarity correlation is positive is taken to consideration. Among the nearest neighbours whose interest is similar to the current user is considered and the result will be evaluated. The result of the recommender system will display whether the current user like or dislike the the unvisited place that their peer user had already visited and rated that place.

Future Enhancement

A travel recommender system has to take care of many fundamental need of a tourist such as transportation, accommodation, destination climate, photography and restaurants and hotels nearby. The travel recommender system can include all these things as a whole package and help user to find the places of their interest. There by enhancing their experience in taking right decisions accordingly.

In future, this can be improved further by applying flexible technique of collaborative filtering and thereby implementing hybrid approach for the recommendation system. That can further improve the accuracy and user experience of the recommendation system. Many new features can be added to the recommender system model like giving best direction and transportation facility for the destination which user chooses.

References

1. Recommender Systems An Introduction By Dietmar Jannach, Markus Zanker, Alexander Felfernig, and Gerhard Friedrich 2011

2. Web Service Recommendation using PCC based Collaborative Filtering by Ms. Sabanaz Sirajuddin Peerzade Assistant Professor, Dept. of Computer Science and Engineering, ATS's Sanjay Bhokare Group of Institutes, Miraj, Maharashtra, India

3. Improving Collaborative Filtering by Selecting an Effective User Neighborhood for Recommender Systems Sundus Ayyaz1 , Usman Qamar2 Department of Computer Engineering College of E&ME, National University of Sciences & Technology. Rawalpindi, Pakistan

4. https://www.kaggle.com/fayomi/advertising/data

5. https://archive.ics.uci.edu/ml/datasets/Tarvel+Review+Ratings

6. https://www.saedsayad.com/naivebayesian.html

7. https://www.geeksforgeeks.org/