# Intelligent Tourist Recommendation System

**Ikshita Mishra[1], Ankita Sharma[2], Tanuj Deria[3]**

Student, Ramrao Adik Institute of Technology[1,2,3]

**Abstract:** This paper introduces a recommendation system which recommends the tourist location to the users, based on their own input preference stated. Input preference is in terms of point of interest, climate, age and budget. This recommendation system uses a dynamic database for recommending the users. Database is created out of survey conducted. Survey is taken from the diverse travelers where they fill their previous tourist trip point of interest, budget, cuisine, climate etc. By matching the user input preference against the survey data stored in database, the output location is recommended.

**Keywords:** Intelligent Tourist Recommendation System, Database, User Input Preference, Survey Data.

## 1. INTRODUCTION

Now-a-days, people are dependent on internet for all kind of mundane information. Web retrieves location details based on the location stated in the search box. But what if the location based on user defined budget, climate preference, interests, age etc. not yet known. Also, going to tourist agents to get an appropriate choice of trip location are expensive these days. In busy schedule, people tend to prefer internet rather than going to some agent. Thus, this recommendation of location based on user preference is a quick process. The recommendation system provides wide scale of options all over the world with perfect match of budget, climatic conditions, interest, age limitations etc. The data is stored in database is taken in form of a survey from travelers. Travelers are made to fill their past tourist trip data. This survey helps the new user to get fact factual data rather than the summed up one.

The aim of the project is to illustrate the use of recommendation system in suggestion tourism places to the users based on their own preferences. The output is showed in Google Maps along with reviews and images of that location.

The project helps in faster exploring of new places matching ones preferences and needs.

## 2. RECOMMENDATION SYSTEM

A recommendation system is a data filter which filters specific information out of varied data and has extensive use in prediction. Recommending and predicting the output from diverse information and user preference, is job of this system. Various examples of usage of recommender system is Movie success prediction, online book recommendation based on the user preference of genre, online retail recommendation based on the past history of various purchases, recommending the medicine to be taken based of ailment stated. These are few instances of recommendation system. There are two types of recommendation system i.e. Content Based Filtering Recommendation System and Collaborative Based Filtering Recommendation System.
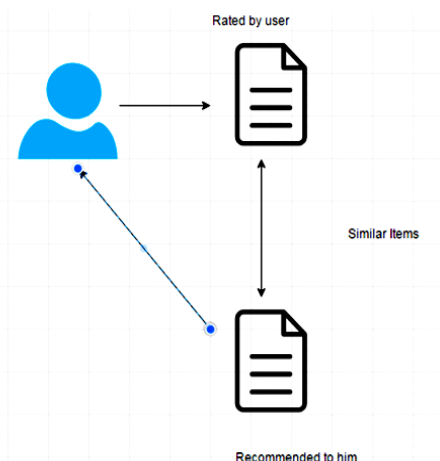
A. CONTENT FILTERING



Fig1. Content based filtering

Content-based systems study the properties of the items recommended. A content based recommender works with information that the user provides, either rating or clicking on a link. For instance, if a Amazon user has ordered many LTE router in the past, then recommend new router classified in the database as having LTE internet speed. Basically, there is an item profile created that is the item set of discrete attributes and features within a system. The recommendation system creates a content-based profile of users based on weighted vector of item properties. The weights denote the significance of each element to the user and can be computed from individually rated content vectors using wide range of methods.

### B.COLLABORATIVE FILTERING
Collaborative filtering is filtering for data or patterns using method concerning collaboration among numerous users, viewpoints, data sources, etc. Applications of this filtering involve several large data sets. The following is an instance of predicting of the user's rating using collaborative filtering. At first, many user rate different movies. Now, an active user amongst them need to rate an unseen movie which is already rated by his peers. The prediction of the rating the unseen movie by the active user is based in his peer ratings for that particular movie. First average of rating is taken, the similarity is calculated based on Pearson's Correlation Coefficient. Then movie rating prediction is calculated for the active agent to rate the unseen movie. For instance, if many peer users LIKE the movie in past, the active user is probable to LIKE the unseen movie.
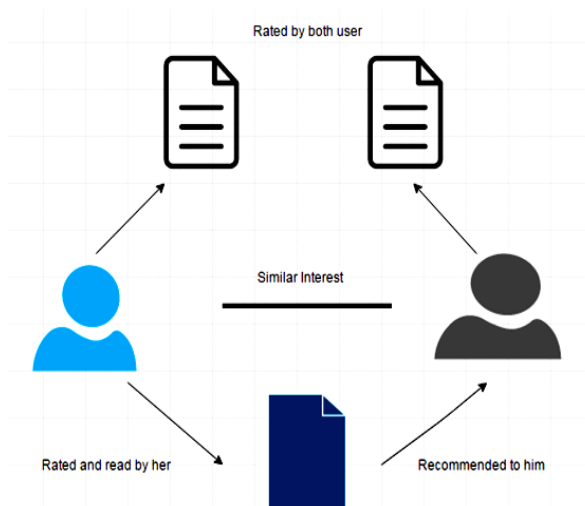


Fig 2. Collaborative based filtering

### 3. ADVANCED TECHINIQUES

#### A.Fuzzy Logic
Fuzzy logic is to approximate a result between 0 and 1. **Fuzzy logic** is many-valued logic in which the truth values are not complete 0 or 1 but in a range. By contrast, in Boolean logic, the truth values of variables are the integer values 0 or 1 only. Fuzzy logic is more or less a partial truth, where the truth value may occupy range between wholly true and wholly false. The coding of fuzzy logic is with help of fuse that is done in JavaScript. Fuse is a Javascript library for fuzzy logic. Javascript Object Notation is used for data storage and retrieval.

#### B.Fuse
Fuse.js is a lightweight JavaScript library which permits you to create a search box on the client-side to search moderately large data sets in a fuzzy way. It allows you to search through a catalog of data by providing a pattern with a maximum length of 32 characters which comprise of misspellings too.
The coding is done in Javascript. The multilevel selection is done through this language. Example, first if a user selects budget, then the locations with various climate are shown for that budget only. Later a second conditioning is done based on climate. Only the required climate under required budget is displayed. Then the selection is done on interest. Such a multilevel classification is done. Then results are displayed.

Steps:
1. Insert coditions.
2. Set options:
a. Include score

b. Include matches
c. Sort
d. Tokenize
e. Minimum character length
3.    Search!

Example :

```
    1. INSERT:
var books = [{
  'title': " The war",
  'author': {
    'name': 'John',
    'tags': [{
      value: 'American'
    }]
  }
},{
  'title': 'The Artist',
  'author': {
    'name': 'Steve',
    'tags': [{
      value: 'English'
    }]
  }
}]

    2.OPTIONS:
var options = {
  keys: ['author.tags.value'],
};
var fuse = new Fuse(books, options)

fuse.search('engsh')

    3.SEARCH RESULT:

[{
  "title": "The Artist",
  "author": {
    "tags": [{
      "value": "English"
    }]
  }
}]
```

C. Lodash

Lodash is a Javascript library that is useful for programmers to write more concise and easier, in order to maintain Javascript. Loadash is a utility class is a class that defines a set of methods that perform common and most often re-used functions. It is generally broken down into various areas:
- Utilities - for simplifying common programming tasks such as determining type and simplifying math operations.
- Function - simplifying binding, decorating, constraining, throttling, currying, and changing the pointer.
- String - performing basic string operations, such as trimming, converting to uppercase etc.
- Array - creating, then splitting , then combining, modifying, and compressing.
- Collection – iterating then sorting, filtering, splitting, and building
- Object – accessing, defaults, extending, merging and transforming
- Seq – chaining, then wrapping, filtering and testing.

Few methods of lodash :
- _times : To loop for N times
- Map: To loop through a collection and return a deeply-nested property from each item
- _uniqueID: To create an array of N size and populate them with unique values of the same prefix
- _.clonedeep: To Deep-clone Javascript object
- _.random: To get Random Number between a range
- _.assign: To extend object
- _.omit: To remove properties from object

D. Open Weather Map API
Current weather data:
We can access current weather data for any place on Earth including over 200,000 cities! Current weather is often restructured based on global models and data from more than 40,000 weather stations. Information is available in JSON, XML, or HTML format.

Call current weather data for one location:

By city name:

Description:

User can call by city name or city name and country code. API responds with a list of outcome that match a searching word. There is a likelihood to obtain a central district of the city/town with its own parameters (geographic coordinates/id/name) in API response.

E. JavaScript Object Notation

Parameters:

- Coord
  o       coord.lon: City geo location, longitude
  o       coord.lat: City geo location, latitude
- weather (more info Weather condition codes)
  o       weather.id: Weather condition id
  o       weather.main : Group of weather parameters (Rain, Snow, Extreme etc.)
  o       weather.description : Weather condition within the group
  o       weather.icon : Weather icon id
- main
  o       main.temp Temperature: Unit Default: Kelvin, Metric: Celsius, Imperial: Fahrenheit.
  o       main.pressure : Atmospheric pressure (on the sea level, if there is no sea_level or grnd_level data), hPa
  o       main.humidity : Humidity, %
  o       main.temp_min :Minimum temperature at the moment. Unit Default: Kelvin, Imperial: Fahrenheit , Metric: Celsius.
  o       main.temp_max : Maximum temperature at the moment. Unit Default: Kelvin, Imperial: Fahrenheit, Metric: Celsius.
  o       main.sea_level : Atmospheric pressure on the sea level, hPa

## 4. IMPLEMENTATION

A. Survey:

Firstly, a survey is conducted amongst the customers based on details about the previous trip. The trip location name, point of interest, expense, climate and their age is taken into consideration. In this way, dataset for the database is created. The dataset keeps on incrementing and modifying dynamically based on survey. Refer Fig.3. for survey conducted and database creation.
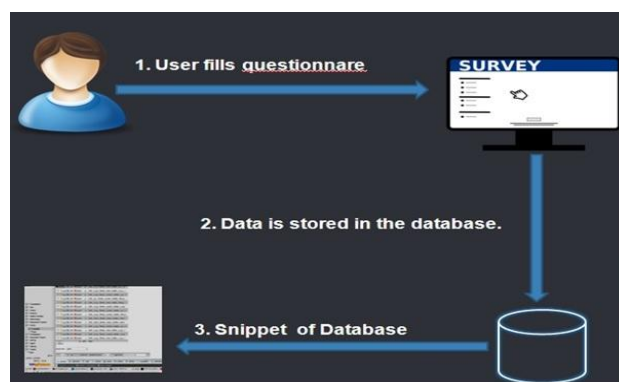


Fig.3 Database Creation from survey.

B. User Section

A login page will appear initially and the user has to login in order to continue. If the user is new he/she will have to fill the registration form (username, password, email, firstname, lastname, etc.) .These details will be submitted to an online database. The second part is about the questionnare which is filled by the users in order to predict their trip locations. A questionnaire will be displayed where one has to enter  user  self requirement based on interests, budget, climate preference. The criteria stated is matched against the data in database and thus the their output location is displayed. Clicking on each result will provide the user with the description of locations along with the pictures. The user will also have the option to view the exact location of the spot on Google Map.  Refer Fig. 4.
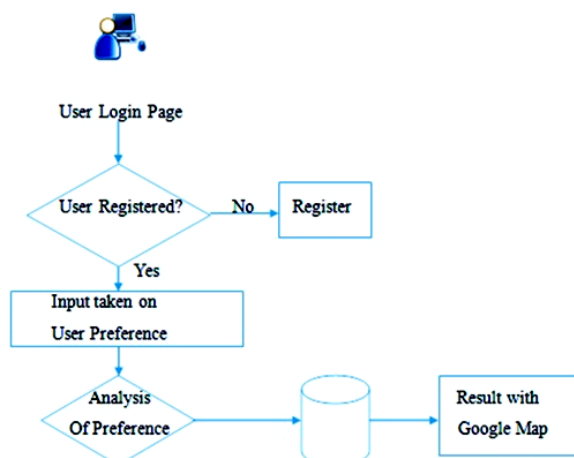
Fig.4. User input and output

### C. Google Maps:

The user will also have the option to view the exact location of the spot on Google Map. On clicking the markers or pointers of points of interests the user will be provided with the place details such as images , reviews, ratings, address, current weather condition. User can also find point of interest in your near by location direction.
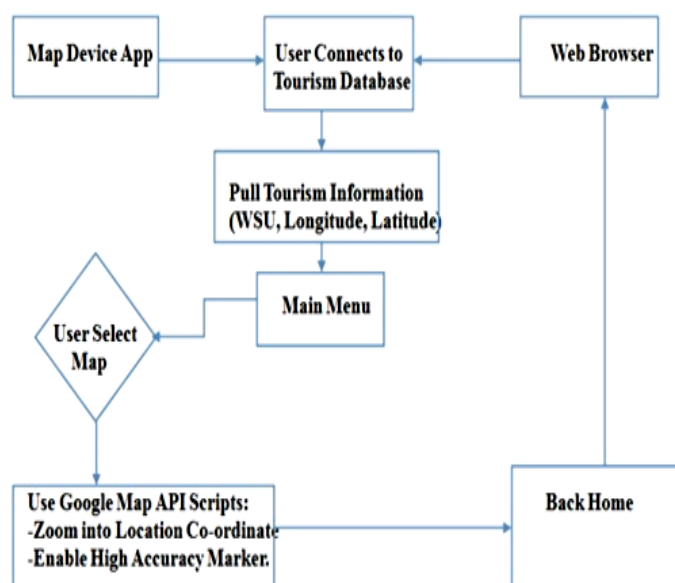


Fig.5 Google Maps Working.

### D. Final System:

• Using Fuzzy Logic, Fuse and Loadash methodology, recommendation is done. Survey is taken by customers based on their previous trip and age, budget, climate is taken into consideration .Questionnaire will be displayed where one has to enter the name, age, gender, budget, climate preference. Dataset is created. User registers and adds his input preference for finding tourist location.

• Fuzzy Logic are used for various combinations of requirement with help of IF-THEN rules. Fuse and lodash are libraries are used to find similar records from the database. The coding is done in JSON for multilevel conditions. Similar records will henceforth be returned to the map page.

• Map page using Google API displays the recommended location and point of interest. Pointers will be provided to display the location details such as images, reviews , ratings, address , current weather

• condition. A survey form in which one will have to enter a place he has visited previously along with budget, climate and point of interest. This data will be added to our database and recommendations will improve with each survey.
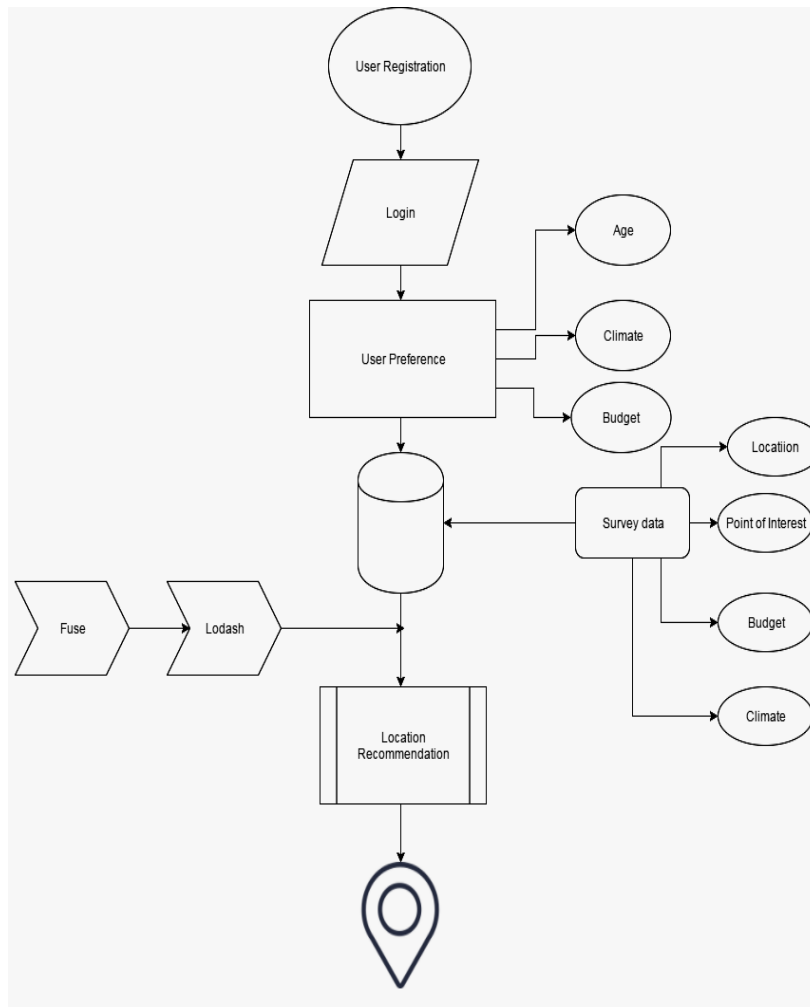
Fig. 6. Final System Working

Example:

Following is the sample dataset collected by the survey:

| ID | AGE | GENDER | BUDGET | CLIMATE | PLACE | CATEGORY |
|---|---|---|---|---|---|---|
| 1 | 10-30 | Female | Upto 10k | Snowy | Shimla | Shopping |
| 2 | 31-50 | Male | 11-40 | Sunny | Thailand | Nightlife |
| 3 | 51-75 | Female | 41-75 | Windy | Spain | Wildlife |
| 4 | 10-30 | Male | Upto 10k | Sunny | Goa | Nightlife |
| 5 | 76 above | Female | Upto 10k | Cloudy | Mumbai | Worship |
| 6 | 31-75 | Female | 41-75 | Snowy | Switzerland | Shopping |
| 7 | 10-30 | Male | Upto 10k | Sunny | Goa | Beaches |
| 8 | 31-75 | Male | Upto 10k | Windy | Mumbai | Museums |
| 9 | 10-30 | Female | 11-40 | Cloudy | Andaman | Beaches |
| 10 | 10-30 | Male | Upto 10k | Sunny | Delhi | Museums |

Fig. 7. Example of Dataset.

Filtering done based on user preference using fuzzy logic.
User criteria input given based on age, gender, climate, budget. Following is snippet of the output obtained.

Example 1:

AGE: 10-30

GENDER: MALE

BUDGET: Upto 10k

CLIMATE: SUNNY

RECOMMENDATION OUTPUT:-

| PLACE | CATEGORY |
|-------|----------|
| GOA | BEACHES, NIGHTLIFE |
| DELHI | MUSEUMS |

Example 2:

AGE: 10-30

GENDER: FEMALE

BUDGET: 41-75

CLIMATE: SUNNY

RECOMMENDATION OUTPUT:-

| PLACE | CATEGORY |
|-------|----------|
| SPAIN | WILDLIFE |
| SWITZERLAND | SHOPPING |

Output displayed on maps:



Fig. 8. Output Displayed.

## 5. ADVANTAGES

Often people get confused while planning for a tour. First of all they hardly know places where they can spend a good time. Further there comes money constraint. This system solves all problems of the client just by asking some questions. And it provides best options that fit into users requirements along with the place details and facilities. It saves their money and time in finding and consulting a travel agency where they charge more. Information at one click

## 6. FUTURE SCOPE

Database Enhancement: As we have limited dataset just by the present survey, our aim is to improve the dataset by conducting many more surveys with variety of choice preference which would enhance our database leading to better recommendation. This is nothing but dynamically improving the dataset by adding couple more of trip details.

## 7. LIMITATION

Initially the options are limited. What if the user needs some out of the box options for choosing preference. The fixed options my limit the applicability.

## 8. CONCLUSION

User preference are the input and output is the location recommendation on Google Maps. Using various advanced techniques such as fuzzy logic, fuse.js, lodash , the implementation of intelligent tourist system is done. Collaborative filtering is used were many people trip details are matched against the required criteria of user. The intelligent tourist system is faster process to display the location ratings and review of that location.

## REFERENCES

[1]  Intelligent Tourist Attractions Recommendation System Based on Cases by Jingyuan Yang, Yonglin Tan, Qian Han.
[2]  Intelligent Search in E-Tourism Services Using Recommendation System:Perfect Guide for Tourist By Juhi Agarwal , Nishkarsh Sharma, Pratik Kumar, Vishesh Parshav .
[3]  Lodash: Javascript functions  by Colin Toh.
[4]  Development and Application of Intelligent Tour Guide System in Mobile Terminal by Shan Li.
[5]  Collaborative Filtering Recommender Systems By Michael D. Ekstrand, John T. Riedl and Joseph A. Konstan