

INTELLIGENT AGENT FOR TOURISM RECOMMENDATION SYSTEM

Jay Borade¹, Shraddha Makwana², Puja Gupta³, Sayalee Lanjewar⁴

¹Assistant Professor Information Technology at Fr CRCE, Mumbai University

²Final year Information Technology student at Fr CRCE, Mumbai University

³Final year Information Technology student at Fr CRCE, Mumbai University

⁴Final year Information Technology student at Fr CRCE, Mumbai University

Abstract

The paper proposes recommendation system for places in tourism domain like food, hotels and travel places and also consists of methods and algorithms for generating such recommender system. Tourism is one of the most important and dynamic industry which needs to accommodate the daily changing requirements of the user. It is one of the most interesting research areas and hence requires a strong recommendation system. In order to find places based on user's criteria in traditional manner would be much ambiguous as it would cost user to find places in separate systems or sometime in order to find such places user has to walk and find information related to that place, but when we use machine learning algorithms and data mining technique to integrate food, hotel and travel based recommendation at one single place it will improve performance in timely manner as well as will make it convenient for the user. Proposed System currently works on data sets of around 7000 records and suggests user from that data. Further system can be expanded to even large data sets. The system shows how qualitative approach can be used with minimal user interface. It gives around more than 80% appropriate results when tested at various locations and it can be further improved with more real time data sets. The paper provides the architecture of our system with implementation details and results. It also explains the flow of the proposed system which covers complete process to be followed, from user login and filling the food survey form to getting different categories of recommendations.

Keywords: - Recommendation, algorithms, restaurants, tourism, django

1. INTRODUCTION

Recommender systems apply data mining techniques and prediction algorithms to predict users' interest on information, products and services among the tremendous amount of available items. [1]. Tourism is one of the most important and dynamic industry which needs to accommodate the daily changing requirements of the user. It is one of the most interesting research areas and hence requires a strong recommendation system. Previously used traditional recommendation methods had their own limitations. In practice, many commercial recommender systems are based on large datasets and as the numbers of users and items grow, traditional recommendation algorithms suffer serious scalability problems and another problem associated with it is "The Cold-Start problem" which occurs due to relatively little information about each user, which results in an inability to draw inferences to recommend items to users [2]. Hence, there is a need of a system that can reduce the challenges associated with previous methods. Use of Machine Learning algorithms in recommendation system overcomes its limitations because

the algorithms here can learn and improve themselves by studying high volumes of available data.[3]

The proposed system uses algorithms like clustering, classification and K-Nearest Neighbor which are part of model based recommendation techniques.

2. OVERVIEW OF PROPOSED SYSTEM

Currently available system largely provides various recommendations but under separate applications. So, the objective of proposed system is to provide a recommendation in domain like food, hotel and travel with various sets of options like time based, nearby places, rating based, user personalized suggestions, etc. at one single place which avoid referring multiple applications and enhances user experience.

The main objective of the proposed system is to provide all the recommendations and guides about the nearby places available.

2.1 Architecture of Proposed System

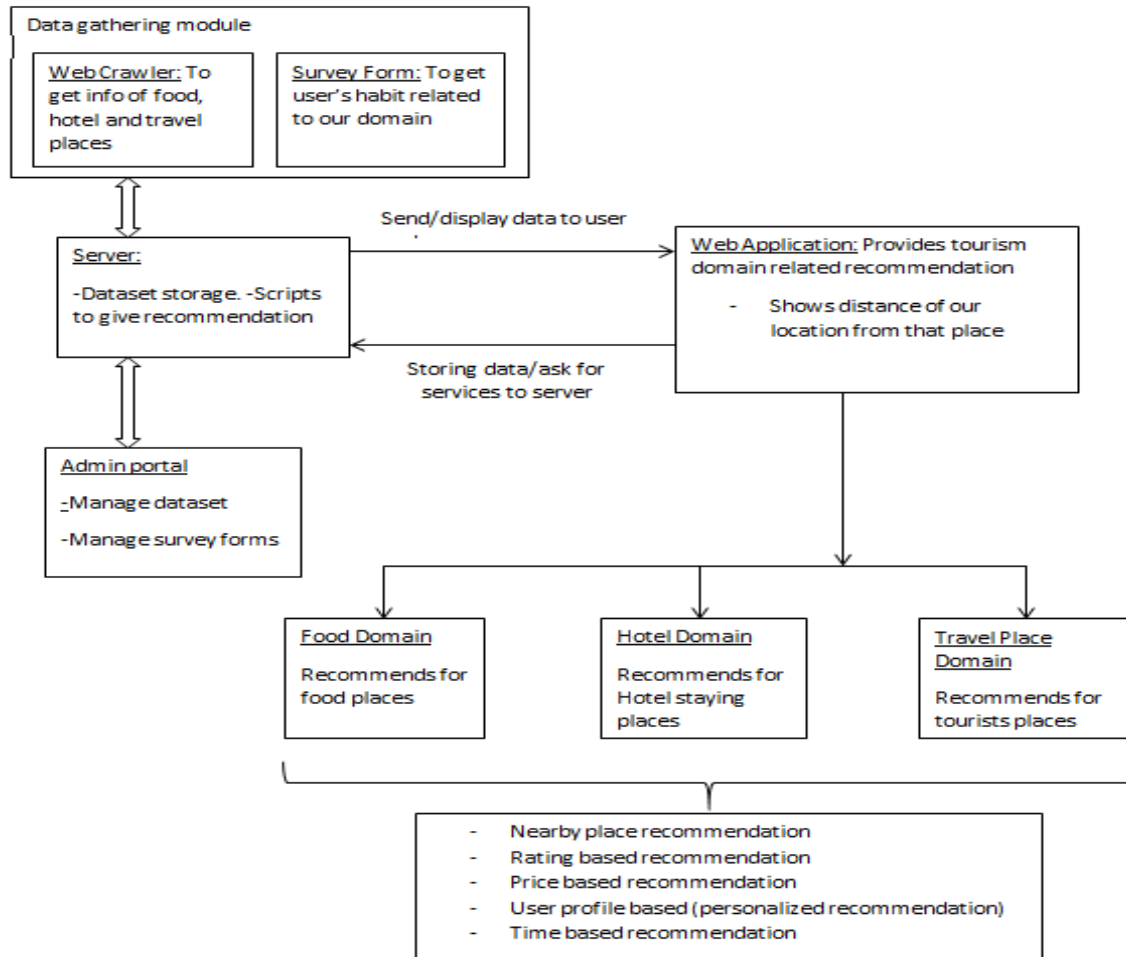


Fig – 1: Architecture of Proposed Recommendation System

It consists of modules of proposed systems and how they communicate with each other in order to understand it more deeply it is necessary to know what these modules does which is given as follows –

These are the different modules of the Proposed System:

- **Data Gathering module:** This module is responsible for gathering of data through internet and survey. For the objective of gathering data, the system uses different Web data crawlers based on different categories. These data crawlers would gather information from internet. The other way of collecting data is through survey when new user get registered. For example here in order to make our prototype here we are getting data from zomato crawler. This module is also associated with data cleaning part where we bring our datasets into the format which machine learning algorithm needs
- **Server:** This module takes care of the storage of the data collected through survey and internet. It is also responsible for the scripts to give recommendation on various criteria. The Server will communicate by sending/displaying the data to user on requests.

- **Web Application:** It provides tourism domain related recommendation to the user. The web application is basically “All in one recommendation guide”. The module will request for services to the Server. The module provides information related to three major domains. Each domain provides recommendation based on specific parameter such as price, rating, nearest distance, time based and user’s preference based.
- **Admin Portal:** The module is responsible for managing the datasets and handling information related to user profile.

2.2 Environmental Set Up

Whenever we design any data product it is important to decide technology stack that has to be used. Here we are building web based travel recommendation system using python technologies such as Django and pandas. For user interface we are using HTML, CSS as well as Bootstrap.

Django development environment needs an installation and setting up Python, Django, and a Database. Since Django deals with web application, it is necessary to do server setup

as well. Following are steps how set up the environment for the proposed application –

1. Installing Python – Python is a powerful high-level open source programming language that is available for multiple platforms. It supports object-oriented programming[4] The core Python distribution is open source and is available for multiple platforms, including Windows, Linux/Unix, and Mac OS X. The default CPython implementation, as well as the standard libraries and documentation, are available free of charge from www.python.org, and are managed by the Python Software Foundation, a nonprofit body.[5]
2. Installing Django – Django can be installed via any of the following options
 - Install an official release. This is the best approach for most users.
 - Install a version of Django provided by your operating system distribution.
 - Install the latest development version. This option is for enthusiasts who want the latest-and-greatest features and aren't afraid of running brand new code. You might encounter new bugs in the development version, but reporting them helps the development of Django. Also, releases of third-party packages are less likely to be compatible with the development version than with the latest stable release.[6]

To verify that Django can be seen by Python, type **python** from your shell. Then at the Python prompt, try to import Django[6]

```
>> import django
>> print(django.get_version())
```

3. Database setup - Django supports several major database engines and we can set up any of them based on our requirement. Such databases are MySQL, PostgreSQL, SQLite3, Oracle, MongoDB, GoogleAppEngineDatastore, etc. We need to configure such databases in settings.py file. In the proposed system we are using SQLite3 database.
4. Web server setup - Django comes with a lightweight web server for developing and testing applications and this server is pre-configured to work with Django, and more importantly, it restarts whenever you modify the code[7]. In order to run our project on this server we need to type “python manage.py runserver” command on command prompt.
5. Installing other packages for Python (needed for our system) – These are add-on packages that we have installed using pip of python for doing some specific task for that system.
 - lxml - This package is corresponding module in standard library to get any web page parsed so that we can process it later. Modern python distribution

like Anaconda has will have lxml already compiled and readily installed[8]

- pandas - We are using pandas to load CSV files for our datasets loads since it is robust IO tools for reading and writing pandas objects to CSV files. It is also 3-dimensional Panel data structure for manipulating collections of DataFrame objects[9]
- geocode - We use this package in order to fetch current location of user in terms of latitude and longitude.
- sklearn - We are using these libraries in order to implement clustering, classification and K-Nearest Neighbors algorithms.
- numpy - this is the package used for large multidimensional array and it makes it very easy to perform massive computational function on such arrays and since in our system also we are dealing with such large data it is very important to use library

3. IMPLEMENTATION DETAILS

To design a functional system, it is important to first decide the mind map of the entire system. The flow diagram mentioned in fig. 2 covers complete process of the proposed system, from user login and filling the food survey form, to getting different categories of recommendations.

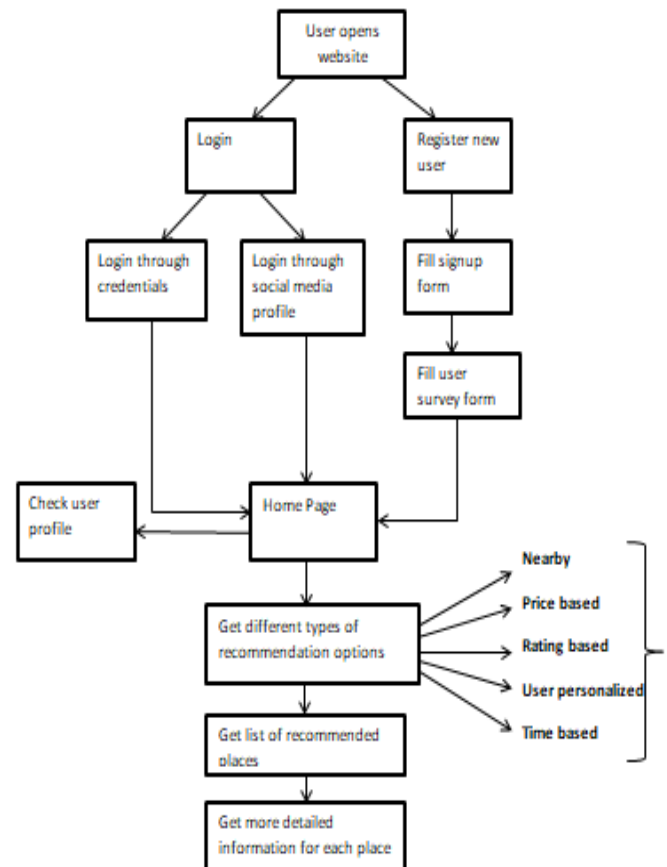


Fig –2: Flow diagram of Proposed Recommendation System

Detailed explanation of the flow of the system is given below.

The user needs to login or signup to access the proposed system. He is also given an option of signing up with his other social media sites. Home Page is the page where the user can access the proposed system and get all the different types of recommendations.

The following diagram shows the various types of recommendations in food domain.

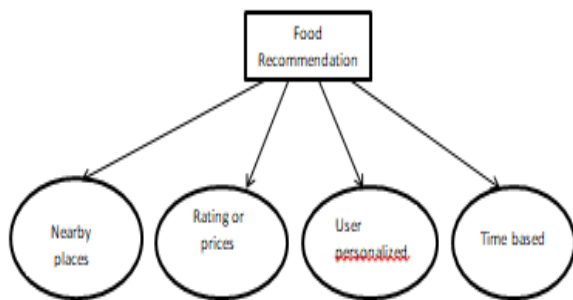


Fig – 3: Types of recommendation in food domain

The home page displays various types of recommendations such as:

- **Nearby places recommendation:** The proposed system uses Nearby recommendation algorithm which provides the user with the details of all the nearby places in its vicinity. The algorithm provides the user with the suggestion of all such nearby places on the basis of the user's current location. The above algorithm makes use of KNN (K Nearest Neighbor) algorithm for fetching all the nearby places. kNN is a case-based learning method, which keeps all the training data for classification. [7]

Nearby places recommendation Algorithm:

- Step1. Start
- Step2. Get current location in terms of latitude and longitude.
- Step3. Get latitude longitude of all restaurants
- Step4. Consider latitude longitude of all restaurants as training set (i.e input parameter of KNN Algorithm)
- Step5. Set nneighbours as input parameter.
- Step6. Apply KNN Algorithm
- Step7. Store nearest neighbor in distances and indices array
- Step9. Stop

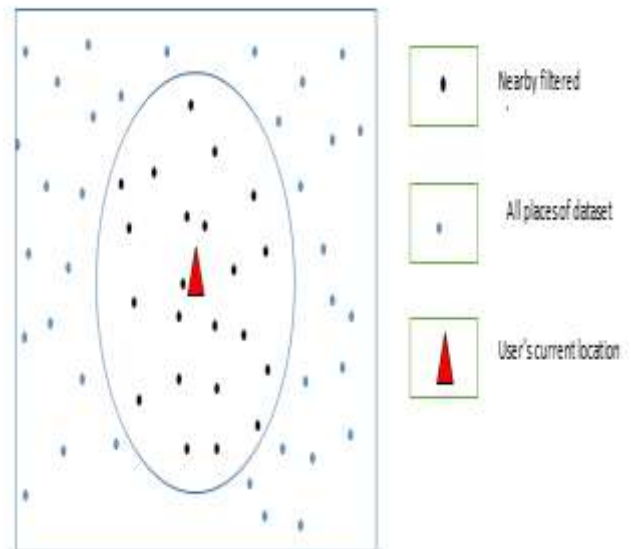


Fig – 4: Cluster formed on the basis of nearby places

- **Price or rating based recommendation:** The rating and price based recommendation algorithms provides the user with the details of all the nearby places by sorting them according to their ratings and prices respectively. The proposed system uses data mining as the places are sorted on the basis of various parameters. For clustering, K means clustering algorithm is used. K means is a unsupervised, non-deterministic, numerical, iterative method of clustering where each cluster is represented by the mean value of objects in the cluster and a set of n object is partitioned into k clusters so that intercluster similarity is low and intracluster similarity is high [10].

Rating/Price based Recommendation Algorithm:

1. Start
2. Find nearby restaurants using above mentioned nearby algorithm.
3. Get cuisine from the user
4. Filter nearby restaurants based on cuisine
5. After that make clusters of above remaining restaurants
6. Apply k-means algorithm and give input parameters as latitude longitude of above filtered restaurants
7. Sort clusters based on distance from the user
8. For each cluster, Sort that cluster based on rating/price.
- End for
9. Stop

Diagrammatic representation

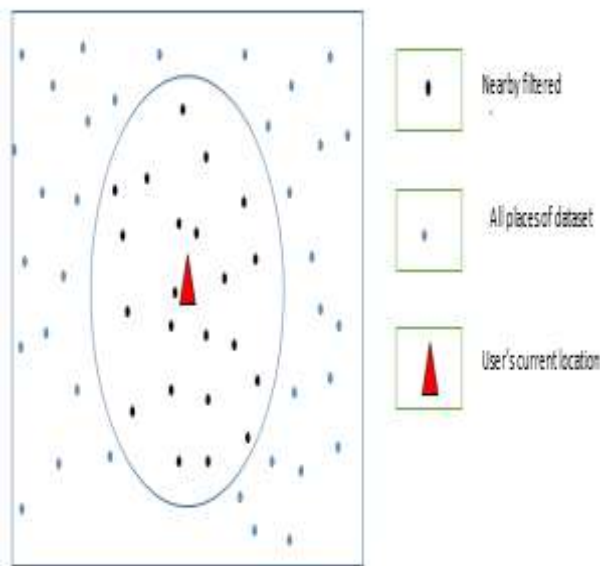


Fig – 5: Clusters formed on the basis of nearby places

After the clusters of nearby places are formed, further clusters are formed by sorting the places on the basis of their ratings or prices. And the cluster closest to the user is recommended to the user.

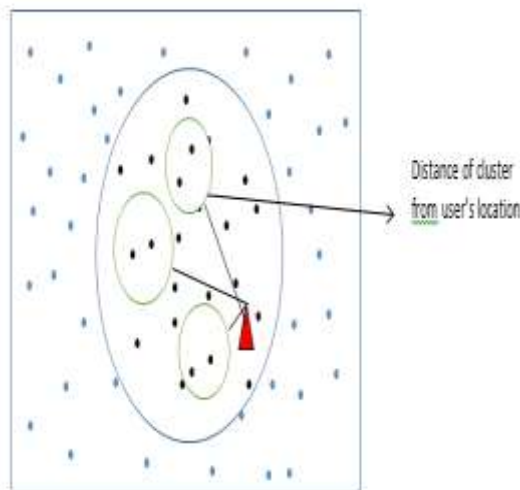


Fig – 6: Clusters formed on the basis of ratings or price

- **User personalized recommendation:** The user personalized algorithm provides the recommendation to the user on the basis of the recorded interests of the user. The classification algorithm is used here to provide better recommendation using decision tree. In decision analysis, a decision tree can be used to visually and explicitly represent decisions and decision making which

uses a tree-like model of decisions and is a commonly used tool in data mining for deriving a strategy to reach a particular goal in machine learning[11]. The decision tree algorithm for the system uses gini index which is a measure of statistical dispersion intended to represent the income or wealth distribution of a nation's residents, and is the most commonly used measurement of inequality [12].

$$G = \frac{\sum_{i=1}^n \sum_{j=1}^n |x_i - x_j|}{2 \sum_{i=1}^n \sum_{j=1}^n x_j} = \frac{\sum_{i=1}^n \sum_{j=1}^n |x_i - x_j|}{2n \sum_{i=1}^n x_i}$$

Use personalized based Recommendation Algorithm:

1. Start
2. Find nearby restaurants
3. Filter them based on cuisine
4. Make training and testing data set
5. Get user's detail
6. Apply decision tree using gini index
7. Classify for user's detail and find restaurant
8. Stop

- **Timing based recommendation:** Timing based recommendation algorithm provides recommendation to the user depending upon the time. For eg. A user will be given suggestion for breakfast in the morning, whereas dinner at night. Data mining is done on available dataset. This is done in order to get appropriate results like membership, branching and loop constructs on large datasets.

Time based Recommendation Algorithm:

1. Get current time and day
2. From timing.csv select all restaurant which are opened currently(rid_current_opened)
3. From timingcuisine.csv select all restaurant for which cuisine falls under current time range(rid_current_cuisine)
4. Select common rid from rid_current_opened and rid_current_cuisine
5. Recommend above common rid.

Restaurant details displayed to the user

When the user clicks on a particular restaurant, the user is redirected to a page where all the details of the restaurant are displayed. It contains all the details such as address, price rating, contact, and other facilities provided by the restaurant.

Restaurant Details	
Name:	Ten One
Address:	883/1JuhuTara Road, Near Melting Pot, Juhu, Mumbai
Avg. Price(for two):	0
Rating:	3.8
More Info:	<ul style="list-style-type: none"> ✘ Home Delivery ✘ Smoking ✓ Alcohol ✓ Wifi ✘ Valet Parking ✘ Rooftop
Email	info@support.com
Phone Number	123-4567-890(Landline) 555-4567-890(Mobile)

Fig – 7: Details of the restaurants displayed

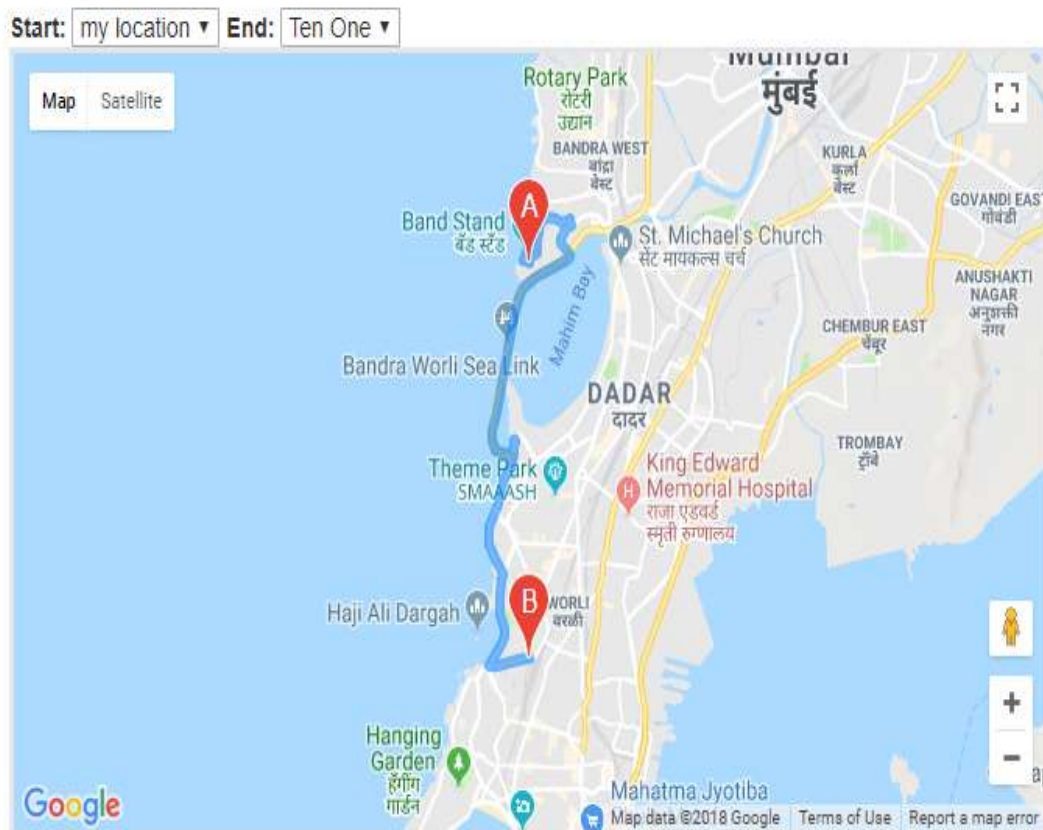


Fig – 8: Route from the user’s location to that particular restaurant

The google map API is used to display the user with the route from its current location to the restaurant recommended to the user which provides it convenience.

he is provided with the exact route from his current location to that particular place. The map displayed on the website shows the respective route to the user.

4. RESULTS AND SUMMARY

The proposed system was tested for various cases. When the user searches for nearby places or any particular restaurants

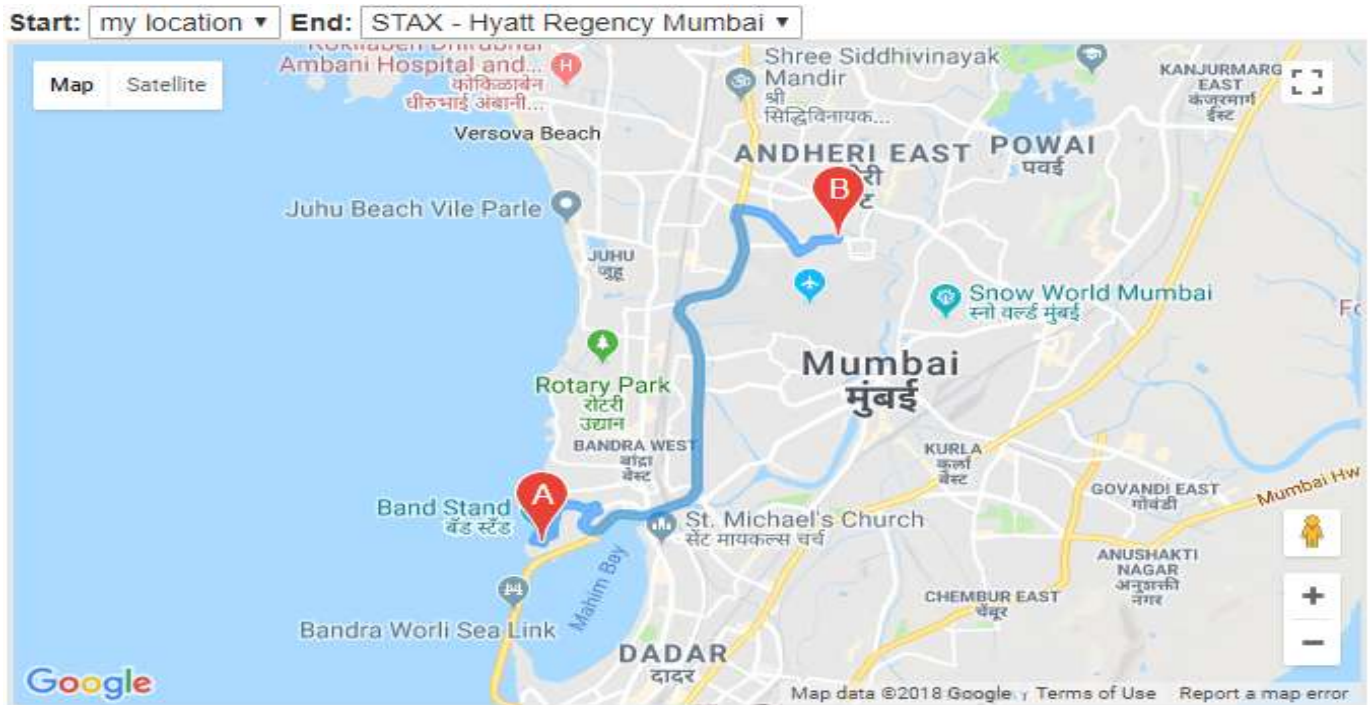


Fig – 9: Route from user’s location to destination

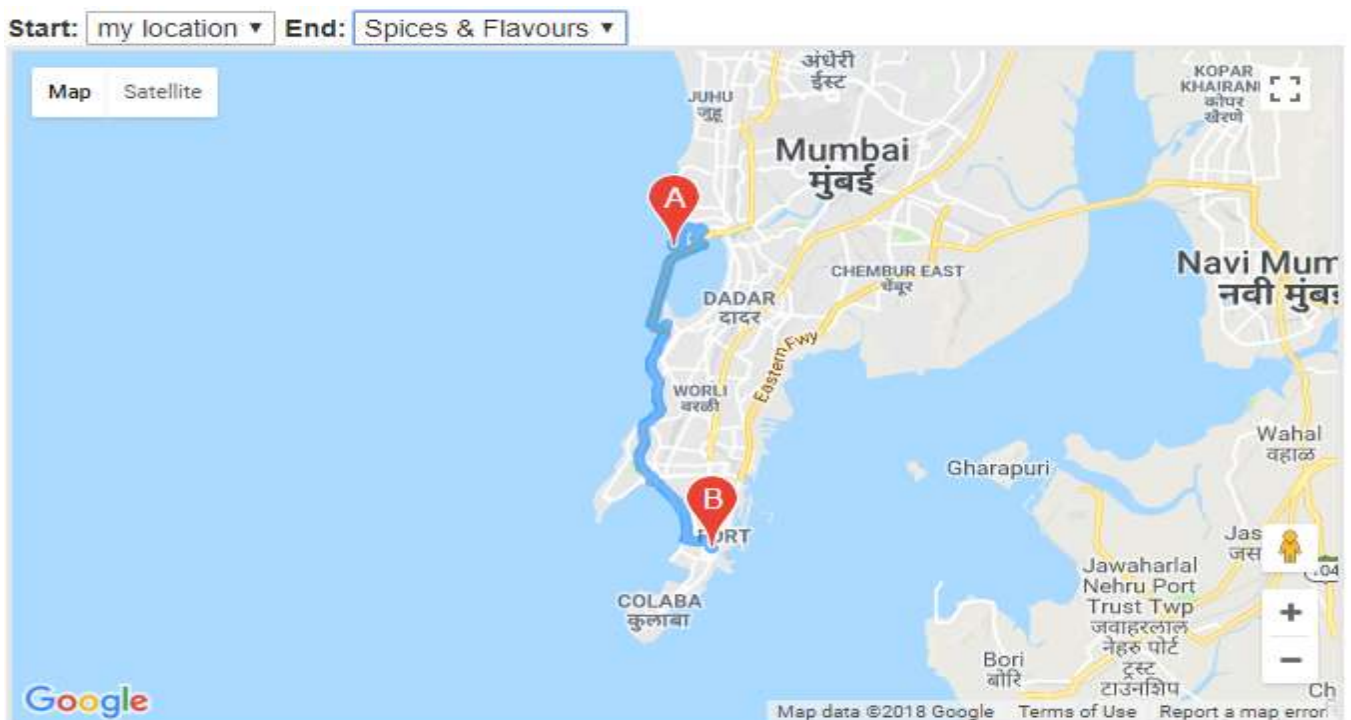


Fig – 10: Route from user’s location to destination

In general, our system performed fairly well, providing relevant information of the places in each experiment that we made. It can be said that the system provided 80% appropriate results.

5. CONCLUSION

The proposed system developed as a project will provide the recommendation system in tourism domain in a one single platform, which will give a better user experience to the user and will also add value to the work of the service provider. The system will provide all the recommendations and guides about the nearby places available. Moreover, it is a platform that accomplishes daily needs of the user at one single place. It also enhances user experience by different category of recommendation based system and also avoids referring multiple applications.

REFERENCES

- [1]. DhohaAlmazro, GhadeerShahatah, Lamia Albdulkarim, Mona Kherees, Romy Martinez, William Nzoukou,"A Survey Paper on Recommender Systems",v4, 24 Dec 2010
- [2]. Mani Madhukar , "Challenges & Limitation in Recommender Systems", Volume 4 ,Issue 3, September 2014
- [3]. ParamitaGhosh , "The Value of Machine Learning: Benefits and Best Practices",May 30, 2017
- [4]. RoselineBilina and Steve Lawford, "Python for Unified Research in Economics and Statistics", Volume 31, Issue 5, 2012
- [5] Almiron, M., Almeida, E., Miranda, M. (2009). The reliability of statistical functions in four software packages freely used in numerical computation. *Brazilian Journal of Probability and Statistics*23:107–119
- [6]. <https://docs.djangoproject.com/en/2.0/intro/install/>
- [7].https://www.tutorialspoint.com/django/django_environment.htm
- [8]. Brandon Rhodes, John Goerzen, "Foundations of Python Network Programming"
- [9]. Wes Mckinney, "pandas: a Foundational Python Library for Data Analysis and Statistics", January 2011.
- [10]. JyotiYadav , Monika Sharma, "A Review of K-mean Algorithm", Volume 4, Issue 7, July 2013
- [11]. <https://towardsdatascience.com/decision-trees-in-machine-learning-641b9c4e8052>
- [12]. https://en.wikipedia.org/wiki/Gini_coefficient