PROJECT TITLE (20pt. bold)

REVIEW OF LITERATURE (14pt. bold)

<u>Machine Intelligence</u>

BACHELOR OF TECHNOLOGY

Department of Computer Science & Engineering

V Semester Section A

SUBMITTED BY

Batch No: 20

Student name 1: Abhishek S Kutre
SRN : PES2UG20CS012

Student name 2: Anish Cherekar
SRN: PES2UG20CS052

Student name 3:Amey Mittal
SRN:PES2UG20CS037

- Research Paper 1 :

  "Mood Based Food Recommendation System"

  (link- https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9545065)

- ***AUTHOR :*** *Manu Gupta, Sriniha Mourila, Sreehasa Kotte, Bhuvana Chandra. K*

ABSTRACT :

The importance of recommender systems cannot be emphasized, considering their pervasive use in many online applications and their ability to diminish many problems associated with over-choice.

The proposed system is completely personalized for users, this system recommends food and available restaurants based on user's current mood.

The dataset from Zomato is taken to locate the restaurants based on location of user. In this system, a website is created which is user-friendly for users where they must register, and login first and then must select their mood. After selecting mood, the user is suggested with food and top-rated restaurants where that food is famous. While suggesting, the best and top 3 are suggested first and then there are some other suggest where customer can select and order the food from their favourite restaurants. Total 9 best restaurants suggested with some quality food to try. Famous food of the top-rated restaurants is displayed below i.e., immediately after the restaurant's name and average cost for two people is also displayed for customers convenience.

TECHNIQUES USED :

- This model is developed using PyCharm, the restaurants are grouped by location using KNN algorithm.
- A html page is created using HTML and JavaScript and with the help of PyCharm we have generate the working model.
- Flask is used to create website which is user friendly.

INTRODUCTION :

The proposed system aims to overcome the demerits of the existing systems by giving food recommendations based on moods and providing interactive interface and personalized recommendations. The restaurants recommended in this model are based on collaborative filtering i.e., through rating, which are grouped by the location of the customer.

This website not only concentrated on moods and ratings, but also on locations of the customer to sort out restaurants in a particular region. The food suggestions provided in this are also based on nutrition, party, cost, delivery of food items etc. are taken while suggesting a food or restaurant for a customer. The rating of a restaurant is done using opinion mining on unstructured data, because the dataset has reviews of customers instead of direct rating of restaurants.

A. Dataset :

In this study, datasets are taken from Kaggle for building the model and for mood-based recommendations for a customer. It includes two datasets, "Zomato" and "food choices" for recommending through rating and mood, respectively.

B. ***Classifier:***

A classifier is an algorithm that automatically categorizes or sorts of data into one or ore "classes." In proposed recommendation system K means clustering classifier is used. By using this techniques, we give data to the design and able to obtain an accurate model.

C. ***Recommendation System :***

A subclass of an information filter system to predict a user's "rating" and "preference" for a particular item is the system that recommends or recommends. Recommendation systems normally utilize collaborative and content-based filtering or other methods such as knowledge-based systems.

1) *Collaborative Filtering*: The technology only produces suggestions for different individuals or things with rating profiles. When pairs of individuals with equal rating histories are found, suggestions are generated from this area. Collaborative filtering methods are classed memorialized and based on models.
2) *Content based filtering*: Content-based filtering algorithms are based on a product description and a profile of consumer preferences. These solutions are most suited when the known data is provided on a single item (name, locale, description, etc.) and are not exposed to the user.

▪ *FLOWCHART :*

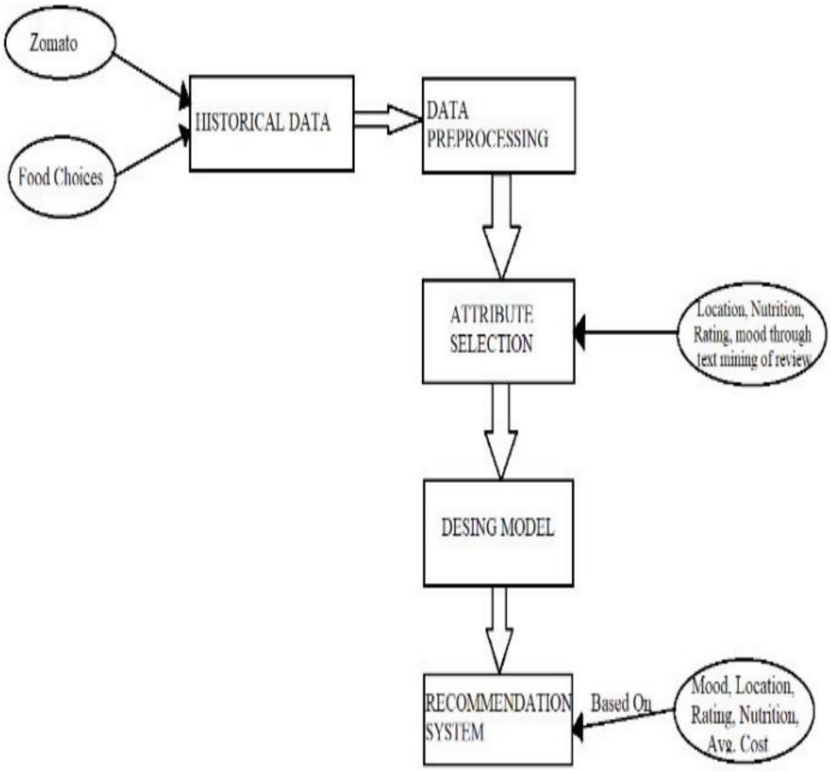


Fig. 1. Flow Chart of Proposed System

- Research Paper 2 :

“Restaurant Recommendation System for User Preference and Services Based on Rating and Amenities ”

(link- https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8862048)

- ***AUTHOR :*** *R.M. Gomathi, P.Ajitha, G. Hari Satya Krishna, I. Harsha Pranay*

- ***ABSTRACT :***

  Recommendation systems are being enforced to offer personalized set of services to the users. They are basically build to produce recommendations or suggestions (like restaurants) that comply with user’s concern and that can be applied to multiple fields. To enhance the quality and service of Recommendation systems and to resolve any issues related to it, various effective techniques linked to data management can be made use of.

- ***TECHNIQUES USED :***

  - Natural language processing (NLP) is one of the machines learning technique to analyse, understand, and derive meaning from human language in a smart and useful way.
  - The evaluation results reveal that the proposed NLP algorithm improves the performance when compared to existing algorithms.
  - The focus of the research work is to offer list of recommended restaurants that is more precise and accessible. The conclusion and results reveal that the suggested approach yields high accuracy.

- ***INTRODUCTION :***

  The current paper proposes a machine learning algorithms to resolve the issue of personalized Restaurant selection relying upon tripadvisor.com search data. The data to be utilized in restaurant recommendation system is being fetched from the Tripadvisor.com website. The facilities provided by the hotel along with user’s comments are being utilized.

The NLP - *Natural Language Processing* is imbibed for examining and tagging all the previous user's comments (whether positive or negative) for every hotel, thereafter computing the overall % of the comments and storing the output.

In the process of Restaurant recommendation, first the user chooses the hotel's features according to his interest and centered on this, the corresponding hotels are fetched. The user reviews are parsed and necessary details (like features and views) are obtained.

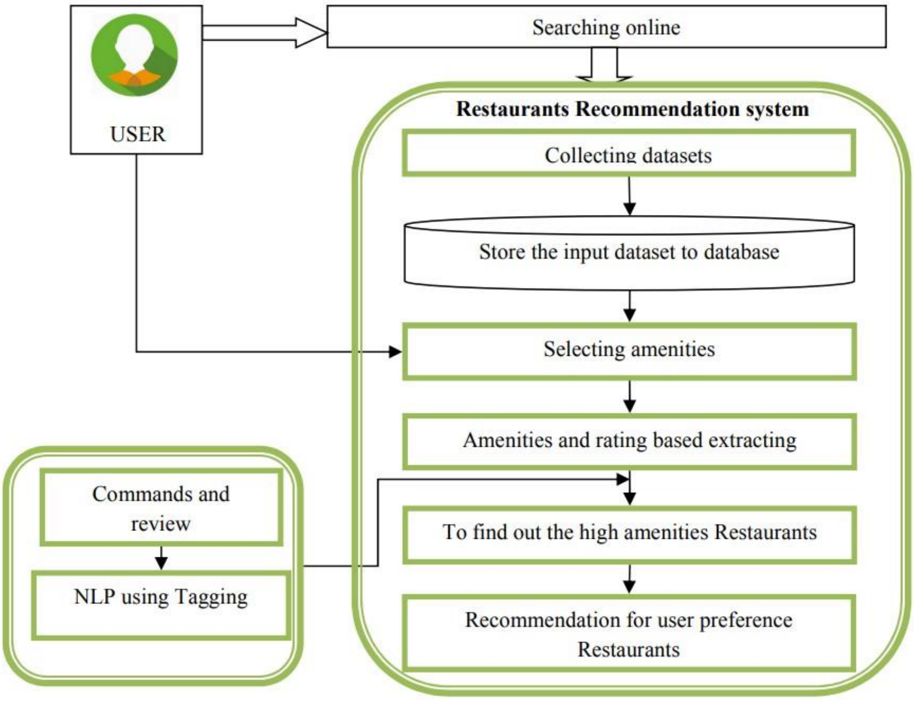
Usually, these reviews (whether positive or negative) hold very significant for the recommender system as they are provided by the users depicting their feelings and views related to a particular restaurant. The summarized rating of a restaurant can be computed in terms of the collective reviews.

Consequently, the features are merged according to user's perspective and a score is generated for every sentence. Then the score of all the sentences are merged to generate a summarized score for a single review and thereafter using the database to store this sentimental outcome.

Using the restaurant recommendation system, the basic hotel amenities are selected by the user and based on this parameter the matching hotels are then populated. The system then uses reviews and comments to analyse the hotel's positive and negative aspects.

Eventually, the highest rated hotel is being recommended to the user by the restaurant recommended system. The proposed sentimental score measure NLP algorithm is used for finding the aspect and sentiments of the user comments.

A. ***Overview :***


USER
Searching online
Restaurants Recommendation system
Collecting datasets
Store the input dataset to database
Selecting amenities
Amenities and rating based extracting
To find out the high amenities Restaurants
Recommendation for user preference Restaurants
Commands and review
NLP using Tagging

B. ***Dataset Collection :***

The data to be utilized is being fetched from the Tripadvisor.com (details like amenities, comments of the users, reviews of the hotel and ratings) to find out the best hotel based on the user gives comments and reviews in particular hotel.

C. *Html Tags :*

The physical layout of the webpage which is presented in HTML is entirely in contrast with visual layout as HTML tags remain concealed.

D. *Amenities Selection :*

This method involves selection from the subset of related features referring the original set so as to lessen the quantity of features involved. The restaurant has different kinds of amenities like fitness centre, High speed Wi-Fi, Room service, Swimming Pool, and valet service.

E. *Amenities Extraction :*

Its a process of mining the most essential data contained within the input data source. The process of feature extraction initializes by analyzing various sorts of restaurants. The amenities extraction is responsible for extracting features from the hotel.

F. *User Rating of Particular Hotel :*

The customer's reviews/feedback forms a helpful source to fetch the restaurant details and user's basic opinion. Based on this rating, the quality of a hotel/restaurant is determined which later forms the input for the sentimental analysis to identify the positive or negative traits related to the restaurant.

G. *Sentimental Analysis :*

In sentiment analysis user's preferences or opinions are being fetched, examined and inspected related to the subjective text. The analysis basically aims to parse the text and determine its polarity (whether neutral, positive or negative).

H. *Sentimental Analysis Using NLP :*

Sentiment Analysis is considered as a NLP approach that focuses on examining user's sentiments and preferences. The NLP processes first tokenize the sentence into words. After tokenization, suffix analysis and prefix analysis is also used for correctly tag each word of sentence. Then use the POS tagging methods to tag each word of sentence of corpus as noun, verb, conjunction, number tag etc and then parsing is done to find the grammatical elements related to each other words and co-reference resolution for pronoun identification of the sentence. Finally, entity recognition is done to understand which word refers to name, person etc.

I. *Recommended System :*

The restaurants are rated and reviewed on the quality and amenities they provide Recommender systems comprehends user's preferences and likings and then deliver personalized information to them.

- Research Paper 3 :

"A Chinese Restaurant Recommendation System Based on Mobile Context-Aware Services"

(link- https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6569074)

- ***AUTHOR :*** *Chung-Hua Chu, Se-Hsien Wu*

- ***ABSTRACT :***

Context information such as user location, time, and user profile, has been popularly applied to analyse user behaviour for many societies. In this paper, we develop a restaurant recommendation system based on mobile context aware services to provide customized information for users. We analyse the service satisfaction ratings of the users to recommend favoured restaurants for them. With mobile context awareness, the proposed framework can substantially enhance the capacity to satisfy the user demands for restaurant recommendations.

- ***INTRODUCTION***

With the rapid growth of mobile technologies, their application on mobile devices can be combined with sensor applications and the development of a variety of Apps, such as the use of the user's location to find nearby businesses applications.

However, it is a fact that most of these Apps only use location Based Services (LBS) to help users find the interested location, but yield a lot of irrelevant data.

There are two types of recommendations on the Internet, content-based filtering and collaborative Filtering. While the former produces results based on the correlation between the content and the user's preferences, the latter derives results based on the correlation between people with similar preferences.

Moreover, context awareness is based on user location, which is variable, meaning that as the surrounding host or access state changes, the provision of information services, such as location-based services, varies accordingly. Therefore, we take into consideration the user's own personal features to improve the filtered results.

A. ***System Architecture :***

The development of the system can be divided into two major parts, namely the mobile device (Client-side) and the server (Server end). The client side is implemented on an iPhone or iPod. The server-side uses an Apache web server and a MySQL database server. The client-side requests data or the relevant parameters via POST such as user surrounding contextual information, personal preferences, and so on. JSON (JavaScript Object Notation) is a data exchange language to respond to the requests.

B. ***Recommendation by Location :***

Such a method of recommending is dependent on location, which is an important factor in the context-aware system. The system will extract the user's location to find out restaurants based on users distance, preventing the situation in which users receive too much restaurant information.

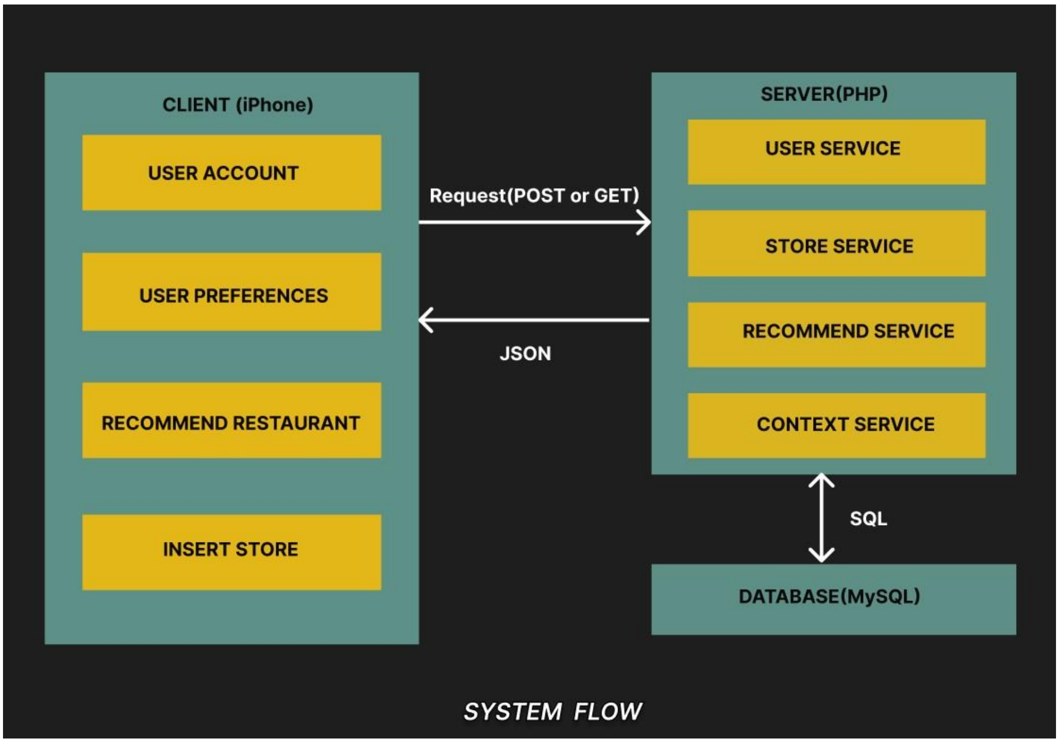
C. ***Recommendation by Context :***

Such a recommendation system collects the user's location, season, and preferences of the type of restaurant. Users can set their preferred time, eating habits, or even hide disliked restaurants, and the system collects the logs of the user's browser for an interactive recommending service. We transform each piece of restaurant information and each user preference factor to vectors, which include the Boolean values. The vector records diet, eating habits, meat and vegetarian dishes in Boolean.

D. ***Recommendation by System :***

The system recommendation combines the above recommended results based on personal preferences and locations to rank these restaurants.

▪ *FLOW DIAGRAM :*



- Research Paper 4

“Restaurant Recommendation System”

INTRODUCTION

Local business review websites such as Yelp and Urbanspoon are a very popular destination for a large number of people for deciding on their eat-outs. Being able to recommend local businesses to users is a functionality that would be a very valuable addition to these site’s functionality. In this paper I aim to build a model that recommends restaurants to users. The way we will model this is by predicting whether a user will have a positive or a negative review for the business. We will restrict to restaurants segment within the business category as recommendation is a very good fit in that system. One way this model could be used in practice is by having an automatic ‘Recommend: Yes/No’ message when a user visits a restaurant’s profile page.

The way the problem is modelled is to be able to predict yes/no for any given restaurant and user. In this work, we will primarily explore the following directions:

1) Optimization algorithms to predict the desired label.
2) Develop features that would help improve the accuracy of this model.

ALGORITHMS USED :

- SVM with RBF Kernel
- Linear SVM
- Logistic regression

PERFORMANCE PARAMETERS :

A. Historical user and business aggregated features The first set of features I implemented were based on intuition that a business is likely to receive ratings correspond to their historical ratings: User-level: Average historical rating from this user, # of reviews Business-level: Average historical rating for this business, # of reviews Missing features – Since historical data for certain users/business can be missing, I circumvented this by using some variations of this feature with default values ranging from min to average to max. Using a default seems to have helped across the board as we give the algorithm a way to treat missing values differently than just zero.

B. User-business category based affinity In order to improve the accuracy further, I decided to implement features that model each user’s personal preference. These features are computed as follows:

Step 1: Compute each user's personal preference on each of the possible business categories and attributes. This is computed as the average rating a user gave to each of the business categories in the historical period. One such feature is avg_rating_for_thaicuisine_for_this_user.

Step 2: In the training and test data, I compute matching features comparing user's preference and the business categories. E.g. the best feature in this category was the average rating for this user averaged over all categories that matched the given business's categories. The intuition behind these features is that user's personal preference on certain categories of restaurants should be a strong signal to whether a user would like a future restaurant.

D. Collaborative Features The publicly available dataset also provided each user's social graph, i.e. the users' friends. Using the intuition that a user's friends' likings are good representatives of a user's likings, I developed the following feature: Given a business and user in the training/test set, average rating for this business from this user's friends in the historical period. As before, I used suitable variations for default values when the feature was missing a value.

▪ *RESULTS :*

1. We see comparable results from the different algorithms that were used although linear SVM was least susceptible to over-fitting and performed marginally better. We achieved a testing accuracy of 69.89% with linear SVM
2. We see a significant improvement from derived features, specifically from using the following: a. Historical average ratings for the business b. Affinity of user to a specific business category c. Collaborative features
3. Increased training data reduced over-fitting, but there's value in weighing training data based on the age of the label. Recent data is more useful in learning than older data.
4. It was important to treat missing feature values differently than zero by providing variations to the model to learn from.
5. At the end, we perform about +5% in accuracy better than the trivial baseline of always predicting yes.

▪ *LIMITATIONS :*

This model only gives us an accuracy of 69.89% using the linear SVM model. Which means that there can be other models used to increase the accuracy.

One more limitation of this model is that this recommendation system can give only positive or negative review.

▪ *REFERENCES :*

[1] Yelp dataset: https://www.yelp.com/academic_dataset

[2] Chih-Chung Chang and Chih-Jen Lin, LIBSVM : a library for support vector machines. ACM Transactions on Intelligent Systems and Technology, 2:27:1--27:27, 2011. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm

Link: http://cs229.stanford.edu/proj2014/Ashish%20Gandhe,Restaurant%20Recommendation%20System.pdf

- ## Research Paper 5 :

"Restaurant Recommendation System in Dhaka City using Machine Learning Approach"

- ### *INTRODUCTION :*

A recommendation system is a model which can predict user interest and suggests the desired item according to the user's preference . Due to the rapid increase of internet users, the recommendation is being used on many platforms nowadays. Many e-commerce websites and other well-known platforms like Netflix, LinkedIn, Facebook, Instagram, and YouTube are using recommendation systems to give better results to their users [2]. Because of rapid development in Bangladesh's economy, as the capital of the country, Dhaka city is developing rapidly. Due to this progress, lots of restaurants have opened their business in Dhaka city. Now there is a situation where there are 8 -10 restaurants in one building. However, the problem is among these restaurants very few are providing good services so it's become a dilemma for food lovers to choose quality restaurants [3]. This research is to try to encounter this problem with a machine learning model to recommend restaurants for the users based on their criteria.

In this research, a new model is proposed where the model deals with two different approaches and merges the best result. Firstly, it takes user input like location, price range, rating, cuisine type. Secondly, it calculates the score based on weight where 90 percent consist of user ratings and 10 percent consists of the price range. After getting the score the model will sort the restaurants based on the score in non-decreasing order and suggest the user top 10 restaurants.

Lastly, by taking users chosen restaurants the model uses another machine learning approach content based filtering, and suggests similar restaurants that users have chosen.

ALGORITHMS USED :

- cosine similarity
- Euclidean distance

PERFORMANCE PARAMETERS :

1) Weighted Based Score Calculation A weighted-based recommendation system is the simplest yet a good way to recommend the best restaurant, it easily calculates a score for every restaurant based on their weighted [12]. The main aim of this method is to calculate an indexing score for every restaurant and sort it in nondecreasing order. Whichever restaurant gets the highest score that means that the restaurant has a good reputation. To calculate the average score we need to consider some features so we choose every restaurant's ratings and several ratings to calculate the average rating scores. average rating R NR RM NRQ NR NRQ (3) Where, R = ratings of a restaurant. NR = number of people who give ratings RM = means of rating column. NRQ = 60% quantile of several ratings. average price P = 3 (4) Where, P = price of that restaurant where 1 = cheap, 2 = medium, 3 = expensive. After calculating the average ratings and average weight for every restaurant we are good to go to calculate the score for every restaurant. We have considered 90% of the average ratings and 10% of the price to calculate our score. The equation is: Score = (average rating * 0.90 + average price * 0.10) (5) After getting the score for every restaurant we can now suggest the best restaurant according to the score. All we need to do is take the user preferences as user input and search them into our database. All the restaurants that will come we index them according to their score and we will get the best restaurant at the top.

2) Content-based Similarity Calculation: After showing the users top restaurants according to their preference, if the user selects any restaurant the model will start to implement its second recommendation system which is content-based filtering. Content-Based filtering is one of the most popular recommendation systems that many well-known companies use to recommend their product to customers. Content-based filtering finds out similarities among the restaurants and which restaurant matches more with the user's chosen restaurant that will recommend first [13]. An example of content-based recommendation is given below for better understanding

REFERENCES :

[1] Nagarnaik, Paritosh, and A. Thomas. "Survey on recommendation system methods." In 2015 2nd International Conference on Electronics and Communication Systems (ICECS), pp. 1603-1608. IEEE, 2015.

Link: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9676197

- Research Paper 6 :

“Food and Restaurant Recommendation System Using Hybrid Filtering Mechanism”

INTRODUCTION :

This paper presents the recommendation system for restaurants and food using a hybrid filtering mechanism. Multiple filtering mechanisms were applied on datasets to recommend food and restaurants for customers. This paper will try to answer user personalization preferences by applying Hybrid mechanisms. The recommendation was based on different customer preference like ratings, top sale, discount, weather condition etc. This paper combines content-based and collaborative based filtering mechanisms to provide the user with full functionalities of the recommender systems. It will adopt a hybrid system from the two mechanisms for effective implementation of the recommendation. To evaluate the performance of the proposed system, machine learning algorithms such as random forest, gradient boosting, decision tree, linear regression and K-Nearest neighbour were applied. The final performance of the model is so promising that it achieved an 83.5% success rate. Model loss and accuracy were also conducted, and the best fitting algorithms were selected. Based on the final result, the random forest algorithm shows significant performance with 0.859 accuracies and 0.1193 loss.

ALGORITHMS USED :

- random forest
- gradient boosting
- decision tree
- linear regression
- K-Nearest neighbour

PERFORMANCE PARAMETERS :

Initial analysis process are performed on data so as to discover patterns, spot irregularities, test hypothesis and check assumptions with the help of summary statistics and graphical representations. Multiple regression multivariate analysis on data fields were performed to see if there is a significant statistical relationship between multiple variables. For each independent variables, multiple regression {x1,x2,x3..xn} were used to input to see the relationships between each of them. Simple regression: Y = d0 + d1 t as well as multiple regression: Y = d0 + d1 t1 + d0 + d1 t2…d0…d1 t were performed on datasets.

The value of performance evaluation for Accuracy, Loss and model evaluation was mentioned between 0 and 1. For Accuracy measurement, the more the value is closer to 1, the more acceptable and accurate the system is. For the loss measurement, the more the value approaches 0, the more accurate the system.

According to the above analysis of algorithms random forest has highest accuracy and lowest loss followed by gradient boosting and decision tree algorithms. Linear regression and decision tree has the highest model evaluation following random forest

REFERENCES :

[1] Keiji Yanai, Takuma Maruyama and Yoshiyuki Kawano, A Cooking Recipe Recommendation System with Visual Recognition of Food Ingredients ,The University of Electro-Communications, Tokyo, Japan2. A Preference-Based Restaurant Recommendation System for Individuals and Groups, page 28, 2014

Link:file:///C:\Users\Dell\Downloads\journal.pdf

- Research Paper 7 :

"Location, Time, and Preference Aware Restaurant Recommendation Method"

(link- https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7860216)

INTRODUCTION :

In this paper, we propose a novel location, time and preference aware restaurant recommendation method using the users' current geospatial location, historical check-in data of the users, and the time of the recommendation request. In the proposed method, users' check-in histories are analysed individually to discover users' visiting trends, food preference trends, and overall popularity of the restaurants. At the same time, each restaurant's operation time and the distance are modelled separately to compute recommendation scores of the restaurants. The recommendation scores are computed by considering four key factors, namely, i) user's preference scores, ii) the distance of the restaurants, iii) the time of a day, and iv) the popularity scores of the restaurants.

▪ *OFFLINE CALCULATIONS :*

1) Modeling Users' Visiting Trends :

The restaurants will experience a gradual change on users' visits with the passage of time. we have n users and m restaurants in a city. An user i visits to restaurants j at time t is represented by vti,j={0,1}. The value of a visit is 1 if user i visits restaurants j at time t. Otherwise the value is 0. If an user i visits to restaurants j at time t, then a weight wti,j is measured using Equation 1. The range of weight wti,j is [0, 1]. The idea is that a latest visit to a restaurant at time t gets higher weight than the earlier visits.

$$w^t_{i,j} = 1 - \frac{1}{1 + e(\frac{T}{2} - \alpha_{v^t_{i,j}})},$$

2) Discovering Users preferences:

Preferred food types of a user play a vital role in choosing a restaurant for having a meal. A user's food tastes change depending on the sessions, and over the period of time. Let us assume that there are p categories of food served in the whole set of m restaurants. A food category k is identified by ζk and each restaurant j∈{ζk}.Mv into a users' food preference matrix Mp as given in Equation

$$M_p = \begin{bmatrix} c_{1,1} & c_{1,2} & \cdots & c_{1,p} \\ c_{2,1} & c_{2,2} & \cdots & c_{2,p} \\ \vdots & \vdots & \ddots & \vdots \\ c_{n,1} & c_{n,2} & \cdots & c_{n,p} \end{bmatrix},$$

3) Discovering Restaurants' Popularity :

A restaurant's popularity is another criteria to be considered for a restaurant recommendation. Restaurants popularity are discovered using Mv. Let a vector h=[h1,h2,…,hn]T is considered as “hub” vector and is used to store each user's experience and a vector a=[a1,a2,…,am]T is considered as “authority” vector and is used to store restaurants' popularity. The mutual reinforcement relationship is computed using

$$\mathbf{a} = M_v \cdot \mathbf{h}$$
$$\mathbf{h} = \mathbf{a}^T \cdot M_v$$

4) Modeling Restaurants' Operation Times :

Awareness of time of operations of the restaurants is very important. The 24 hours of the day is equally divided into 6 bins. Let us assume that σbj represents the total number of check-ins to a restaurant j for bin b, and ψj represents the total number of check-ins for restaurant j.

$$M_t = \begin{bmatrix} f_{1,1} & f_{1,2} & \cdots & f_{1,m} \\ f_{2,1} & f_{2,2} & \cdots & f_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ f_{6,1} & f_{6,2} & \cdots & f_{6,m} \end{bmatrix},$$

where

$$f_{b,j} = \frac{\sigma_j^b}{\psi_j},$$

- ***ONLINE RECOMMENDATIONS :***

1) Modeling Restaurants' Distance :

Usually, a user tends to show interest to visit restaurants with good quality in the nearest proximity. In such case, they normally prefer closer restaurants than the distant ones. In order to do that, all the restaurants j within the user's preferred distance are identified and calculated a distance score dj. The distance score lies in [0, 1]. As a result, restaurants closer to a user gets higher distance score of dj, closer to 1 and distance score decreases gradually as the distance of the restaurants δj increase.

$$d_j = 1 - \frac{1}{1 + e^{(\frac{\lambda_i}{2} - \delta_j)}},$$

where λi is the preferred distance of user i given (λi≥1), and δj is the distance of a restaurant j from the user's current location.

2) Generating Recommendation Scores :

On receiving a recommendation request from an user i, a recommendation score ρj is calculated for each restaurant j using four factors, namely 1) the user's preference score on each restaurant category, ci,k;2) the restaurants' popularity, aj;3) the time-awareness score of each restaurant, fb,j; and 4) the distance score of each restaurant, dj.

$$\rho_j = c_{i,k} \times a_j \times f_{b,j} \times d_j | j \in \{\zeta_k\}.$$

Using the computed matrix, users' visiting trends, users' preferences, and restaurants' popularity have been discovered. Then, the restaurants' operation time has been modelled to understand the activities of a restaurant. The distance of a restaurant has also been modelled to fit in the proposed restaurant. After computing these factors, the recommendation scores are computed.

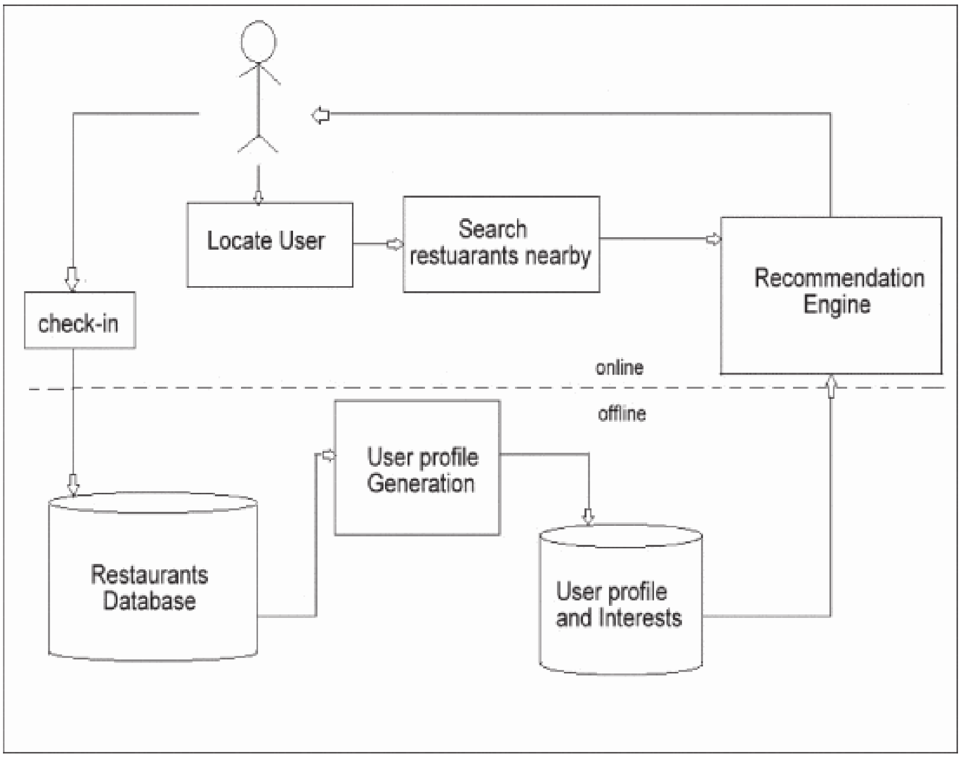
- Research Paper 8 :

“Location based personalized restaurant recommendation system for mobile environments”

(link - https://ieeexplore.ieee.org/document/6637223)

INTRODUCTION:

In this paper, we present a personalized location based restaurant recommendation system integrated in mobile technology. It ubiquitously studies the user's behavioral pattern of visiting restaurant using *Machine Learning*. We would use Foursquare to extract all our data and post the user's visits in it. The system architecture presented in figure 1 can be simply divided into two sections, one which has online activity, and the other which processes data offline. When the user is in motion, i.e., his geo-position changes notably, the system goes online and recommendation module becomes active, retrieving nearby and restaurants and ranking them, based on their properties, according to the scores generated offline. The offline part generally remains in a non-functional mode when the user is stationary The work of the offline system is to generate a user interest profile, using a Machine Learning algorithm, from the data set that keeps getting modified whenever the user checks-in a restaurant.



- ***OFFLINE CALCULATIONS :***

Restaurants are classified as ‘like’ or ‘dislike’, depending on the taste of the user. Here, we make an initial assumption that the user likes all those restaurants if he makes a check-in in Foursquare. Each restaurant would account for n entries, where n is the number of check-ins made. Further, if he likes two restaurants, which one would he favour more? We solve this problem by using Naïve Bayes Classifier algorithm to recognize the factors that the user likes about a restaurant and to what degree does he like them. we have found out that multinomial Bernoulli model would be more efficient in this context. Let the feature vector set be X={category, location} and the set of classification classes be C={like, unlike}. The total weightage of all the features present in the training data is calculated as follows:

$$\mathrm{P}(\mathrm{x}|\{like\}) = \mathrm{N}(\mathrm{x}_i, \{like\}) + \frac{1}{\mathrm{N}(\{\mathrm{like}\})} + \mathrm{V}$$

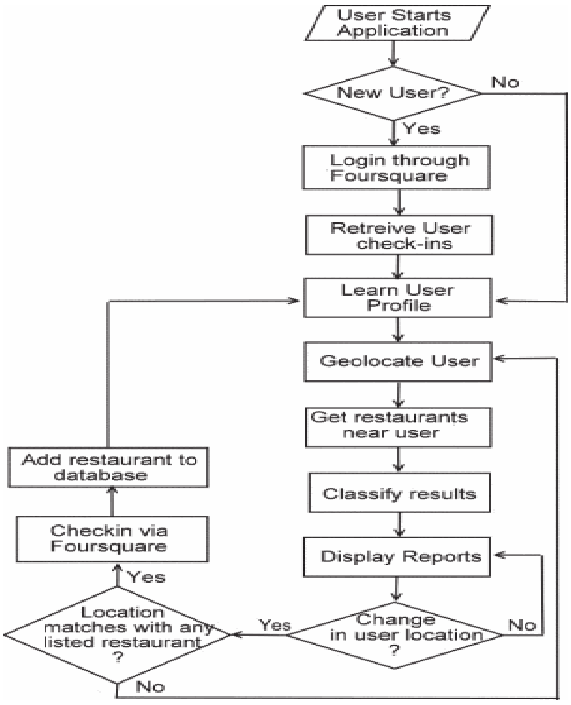
where V is the total number of keywords occurring in the database and N(i) is the number of instances of i in the dataset.

Recalling that the application is always functioning, when it detects a change in the user's geo-located position, it gets his coordinates and finds the restaurants around those coordinates. Each restaurant from the retrieved list is passed onto calculate its probabilistic value. Therefore, the probability that a user likes a restaurant R is

$$\mathrm{P}(\{like\}|\mathrm{R}) = \prod_{i=1}^{n} \mathrm{P}(\mathrm{X}_\mathrm{i}|\{like\})$$

Before proceeding with the calculations, the system would check if any restaurant in the list shares the same name and category, but a different location, with that in the training data (for chain restaurants).

The system would continuously keep on checking the user's location depending on certain events. As seen from the flow chart, there is minimum user intervention after logging in. The user only needs to interact with the application if he needs further details of a restaurant, otherwise, he can just glance at the report.



- Research Paper 9 :

“Worth eat: An intelligent application for restaurant recommendation based on customer preference (Case study: Five types of restaurant in Tangerang Selatan region, Indonesia)”
(link - https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8074654)

- ***ABSTRACT :***

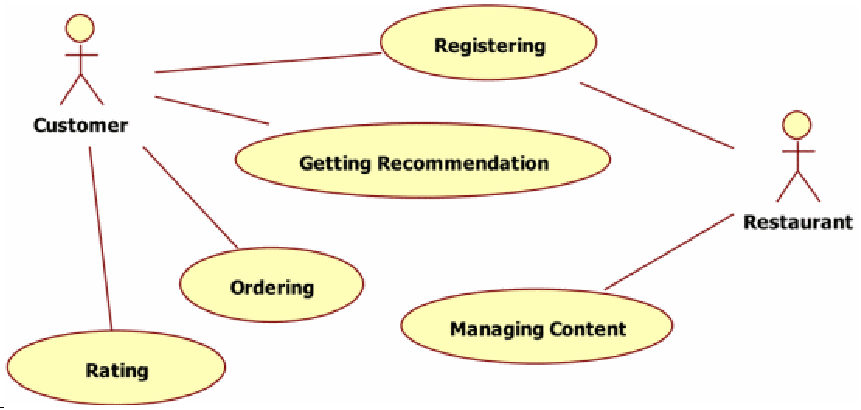
In a metropolitan, to find a restaurant in rush hour is a challenging activity to do. It is going to be easy, when someone accesses an application and it recommend the best restaurant then he/she can visit to. Through using combination between methods fuzzy-logic and bubble-sort that were realized in customer preference case, an intelligent application for restaurant recommendation was technologically constructed. The method fuzzy logic used to parameterize three selected parameters; interest, location, and rating. And, the bubble-sort is a method to optimize the recommendation based on full-factorial concept. The three selected parameters (i.e. distance, rating, and interest) represent three characteristics

illustrating recommender systems; location based, content based, and collaborative filtering respectively; where each parameter illustrates geographic convenience , users attention similarity, and product interest discovery. And the customer's interest of restaurant category is ideally indicating the term “customer preference”.

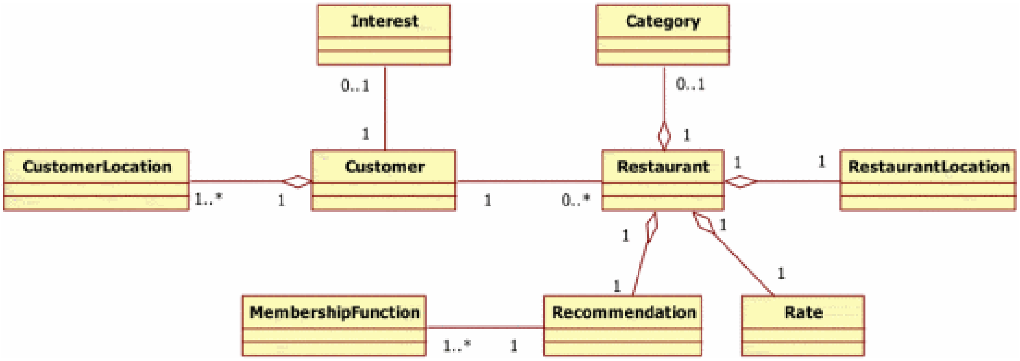
INTRODUCTION:

To analyse and design the constructed application, the method object oriented was used. For recommending the fittest restaurant for customer, the algorithm bubble-sort was operated. The fittest value is obtained based on the calculation of the similarity among the parameters belong to customer and restaurant.

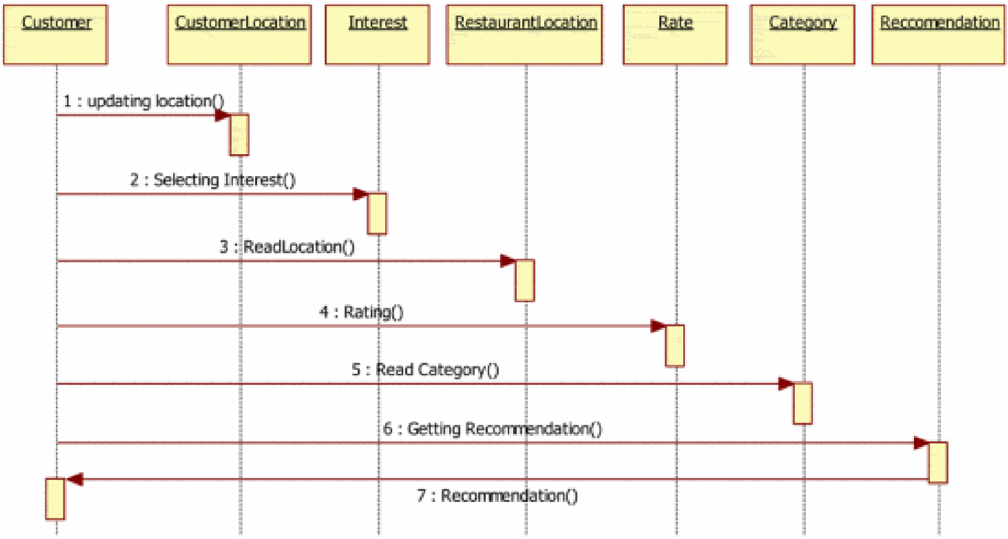
Here, there are two human actors identified; *customer* and *Restaurant*. The actor Customer interconnects to four types of usecase (activity group); *Registering, Getting Recommendation, Ordering, and Rating*. Indeed, the key part of the application is in usecase Getting Recommendation. While, the actor Restaurant mainly interacts to the application for *managing content*, it is facilitated via the usecase Managing Content.



In addition, bubble-sort method that is technically operated to grade the recommendation is implicitly identified in the class Recommendation as an operation. Here, the class *Recommendation* communicates with the class *MembershipFunction* to determine a set of fuzzy language.



Sequentially, the usecase Getting Recommendation is done based on sequence diagram. To conduct the usecase, seven objects are involved. Those objects are used to update the information into or read the information from. The final message (information) coming from this usecase is a restaurant recommendation that customer is able to choose.



An intelligent recommender application was methodically created. It operates three types of parameter (distance, rating, and interest) to look for a similarity value (between customer and restaurant). One of selected parameters (parameter interest) is hypothetically characterizing the “customer preference”. Additionally, the application is able to recommend the fittest restaurant for customer who is looking for a restaurant to select. A combination between methods fuzzy-logic and bubble-sort were officially adopted as main concept in constructing the application.