Tourist Place Recommendation System using Social Networking Data

Akshay Babhulkar[1] Sudarshan Hadapkar[2] Trupati Phutane[3] Neelkamal Bhandari[4] Ganesh Mandade[5]

[1,2,3,4,5]Department of Computer Engineering

[1,2,3,4,5]D Y Patil College of Engineering

Abstract— Recommender system using social networks to promote smart tourism Latest revolution in computing areas such as communication involving networks of people (social networks), intelligent devices, smart mobile computing, and communication devices that will form cyber-physical social systems. So can we make best use of this enormous data sources to proactively recommend different items to users, based on user context. This category of recommendation systems is Push recommendations. We are trying to propose a recommender model, which shall use implicit, local and personal information of the user from the Internet of Things, Social Account, user web browsing history, smart devices etc. where anything will be connected any time. This proposed system shall be pushed to the user, and not of only 1 type. This can be used by the end users to avail the facilities and services, but for which users has to register and Travel and Tour companies will register and upload the services and facilities they offer. So this system can provide vendors new opportunities to get new business and consumers can avail the services and facilities.

Key words: Tourist data, User based Collaborative Filtering, Item based Collaborative Filtering, SVM, Facebook and Twitter API

I. Introduction

The large amount of information are available in the World Wide Web and its number of users have increase in the last decade. Information is mostly used for those users who plan to visit an unknown destination. Information about travel destinations and their resources, such as accommodations, restaurants, museums or events, among others, is commonly searched for tourists in order to plan a trip. However, the list of possibilities offered by Web search engines may be overwhelming. The evaluation of this options is very complex and time consuming for tourists in order to select the one that fits better with their needs.

Personalization techniques aim to provide customized information to users based on their preferences, restrictions or tastes. They are particularly relevant in recommender systems, whose objective is to filter.

irrelevant options and to provide personalized and relevant information to each particular user. In the tourism sector, travel recommender systems try to match the characteristics of tourism and resources or attractions with the user needs. These systems are very useful if they can automatically learn the user's preferences through the analysis of the explicit or implicit feedback. Explicit data may be given by the user in different ways, for instance whenever she specifies her cultural interests by filling in a form. Implicit interests can be entered by the system through the analysis of the behaviour of the user.

In many cases, recommender systems it takes into account the preferences of the tourist as well as analyze the dynamic context of the trip. This is especially useful when tourists are already at the destination and they are willing to use their mobile devices to customize their trips in real time. The context can include aspects like the tourist's location, the time of the visit or the current weather (Lamsfus et al., 2009). Approaches that take context into account can send suggestions proactively, depending on the current state of the tourist. For example, a museum that was planned to be visited today may be too far from the visitor and she may not have enough time to reach it, so the plan scheduled for tomorrow may be changed to include the visit to the museum..

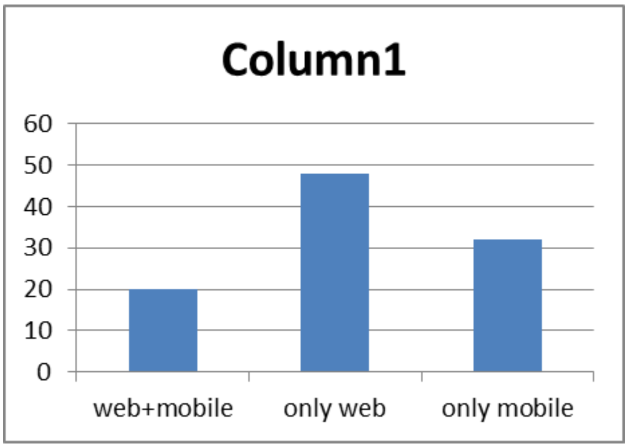
II. Interface

This section analyses the user interfaces of recent tourism recommender systems. Most offer a Web-based interface and/or an interface specifically designed to be used in mobile devices. Table 1 classifies the most relevant e-Tourism recommenders in these two broad categories, and Fig. 1 shows the percentage of surveyed systems in each of them. A Web-based interface is the option chosen by most of the systems, since it permits an easy access from any computer connected to the Web without any kind of downloading, installation and configuration. However, due to the enormous increase in the use of smart phones connected to the Web in the last years, more than half of the reviewed systems have specific interfaces for mobile devices.

Interface	Reference
Web+mobile	(Venkataiaha et al., 2008), (Lamsfus et al., 2009), (Niaraki and Kim, 2009), (Vansteenwegen et al., 2010), (Gavalas and Kenteris, 2011), (Rey-López et al., 2011), (Borràs et al., 2012a), (Ruotsalo et al., 2013), (Umanets et al. 2013)
Only web	(Coelho et al., 2009), (Huang and Bian, 2009), (Lucas et al., 2009), (Lee et al., 2009), (Ruiz-Montiel and Aldana-Montes, 2009), (Jannach et al., 2010), (Mínguez et al., 2010), (Sebastià et al., 2010), (Yang, 2010), (Borràs et al., 2011), (García-Crespo et al., 2011), (Linaza et al., 2011), (Lorenzi et al., 2011), (Luberg et al., 2011), (Montejo-Ráez et al., 2011), (Sebastià et al., 2009) (Garcia et al., 2011), (Wang et al., 2011), (Koceski and Petrevska, 2012), (Gyorodi et al. 2013), (Kurata and Hara 2013), (Lucas et al., 2013), (Savir et al., 2013)
Only mobile	(Castillo et al., 2008), (Ceccaroni et al., 2009), (García-Crespo et al., 2009), (Yu and Chang, 2009), (Ricci et al., 2010), (Martin et al., 2011), (Batet et al., 2012), (Martínez-Santiago et al., 2012), (Noguera et al., 2012), (Braunhofer et al. 2013), (Garcia et al. 2013a), (Meehan et

	al., 2013), (Rojas and Uribe 2013), (Yang and Hwang, 2013)

III. GRAPH



There are some recommender systems that have been designed as desktop applications and do not offer any of the two usual kinds of interfaces (e.g., (Kurata, 2011)). This kind of applications can usually be implemented more quickly than the mobile or Web-based ones; however, they require downloading and installing the program, which is not comfortable to most of the tourists that want to get recommendations as simply as possible without being bothered by technical details.

The following subsections review some approaches based on Web or mobile interfaces.

IV. WEB-BASED RECOMMENDERS

The use of a Web-based interface is the most common option adopted by eTourism recommenders. This kind of interfaces allows tourists to look for information friendly manner. Users normally have a rich interaction with the system using a wide screen allowing displaying a large amount of data extended with maps, images or even high quality videos. Moreover,it permits to interact easily with the computer and move through maps, perform zoom actions, select items or even drag and drop them. This is useful for tourists when they are still in the planning stage of their trips. These Web based applications are usually not designed to be used during the stay since most of the tourists will not have easy access to computers with Internet connection. Although an increasing number of tourists have mobile handsets or tablets with Internet connection, the information-ridden Web pages usually shown cannot be easily read or manipulated on such small screens. In the remainder we comment some interesting features exploited in Web based interfaces to improve the interaction with the users.

(Venkataiaha et al., 2008) report the design of two visualisation systems (called discrete and continuous) for a tourism recommender and compare the interaction of the users in both cases. The former provides a high quantity of information in the screen at the same time, and it was analyzed that users needed too much time and effort to understand it. The latter aggregates all the information into a single video clip that combines the most relevant media content, including text, photographs and videos.

A. Existing System

In existing system two main approaches to sentiment analysis which are lexicon-based approach or machine learning approach. In multiple system which done sentiment analysis either using are lexicon-based approach or machine learning approach.

In existing system for tourist place recommendation system recommend the tourist place with the help of collaborative filtering on tourist static dataset. There are not involving the real time social sites (Twitter and Facebook) user feedback for recommendation process.

B. Draw backs:

For sentiment analysis lexicon-based approach or machine learning approach are used but we don't have any method are available to use both lexicon-based approach and machine learning approach combineExisting recommendation system work on only static datasetNo real time tweets or post data was used for tourist place recommendation

C. Proposed System

To propose a recommender system based on Data acquisition from Tourism dataset which is static, IoT static dataset which store the IoT details of tour places(temperature, humidity, pollution etc.) and extract the Facebook and Twitter data related to tour places. We develop a web application in Java for tourist place recommendation system.

D. Proposed Algorithm

1) Step 1: User enter the search query (like user age, tour category and duration of tour)
2) Step 2: After entering user search query we use the user based collaborative filtering for tour place selection which is apply on tourism static data.
3) Step 3: The completion of user based collaborative filtering we got places with visited count and user the top five place for next step.
4) Step 4: In this step, we fetch the Twitter and Facebook data related to above five selected places. We used API for Twitter and Facebook Data Extraction.
5) Step 5: After extracting Twitter and Facebook data we used item based collaborative filtering and SVM algorithm with NLP technique.
6) Step 6: SVM classify the Facebook post and Twitter Tweets into positive and Negative.
7) Step 7: Compare the User based CF and Item based CF result and SVM result and recommend the places as per the search query.
8) Step 8: In result page we display the recommended places with IoT and Tourist details

E. Advantages of proposed system

The Environment of the IoT is providing much information about the user context.
User shall promptly receive recommendation based on user just real time context.
Based on Recommendations, users shall buy/purchase products or services and hence the
Retail Businesses and Tourism Industry shall ourish get more revenue.

V. ALGORITHMS

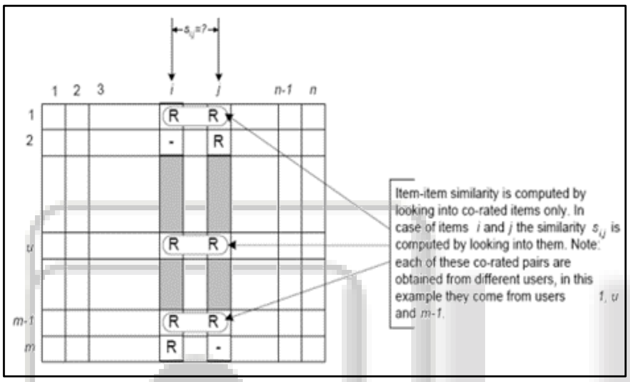
A. *Algorithm 1: Collaborative Filtering*

Using tourism database (we use the user based collaborative filtering for tour place selection which is apply on tourism static data) we find the tourist record which fulfill the user requirement or search query and finally we got multiple tour places.

The completion of user based collaborative filtering we got places with visited count and user the top five place for next step.

Using Facebook and Twitter API, we extract the twitter tweets, likes, re-tweets and Facebook post, comment, likes and other data.

The similarity values between items are measured by observing all the users who have rated both the items. As shown in the diagram below, the similarity between two items is dependent upon the ratings given to the items by users who have rated both of them:



1) *Similarity Measures*

There are number of different mathematical formulations that are used to calculate the similarity between items. As we can see in the formulae below, each formula includes terms summed over the set of common users U.

2) *Cosine-based similarity*

Also it is known as vector-based similarity, this formulation shows two items and their ratings as vectors, and describe the similarity between them as the angle between these vectors:

$$sim(i,j) = \cos(\vec{i}, \vec{j}) = \frac{\vec{i} \cdot \vec{j}}{||\vec{i}||_2 * ||\vec{j}||_2}$$

3) *Pearson (correlation)-based similarity*

It is based on similarity measure and how much the ratings by common users for a pair of items deviate from average ratings for those items:

$$sim(i,j) = \frac{\sum_{u \in U}(R_{u,i} - \bar{R}_i)(R_{u,j} - \bar{R}_j)}{\sqrt{\sum_{u \in U}(R_{u,i} - \bar{R}_i)^2}\sqrt{\sum_{u \in U}(R_{u,j} - \bar{R}_j)^2}}$$

4) *Adjusted cosine similarity*

This similarity measurement is a modified form of vector-based similarity where we take the fact that different users have different ratings schemes; some users might rate items highly in general, and others might give items lower ratings as a preference. To remove this drawback from vector-based similarity, we subtract average ratings for each user from each user's rating for the pair of items in question.

5) *From model to predictions*

$$sim(i,j) = \frac{\sum_{u \in U}(R_{u,i} - \bar{R}_u)(R_{u,j} - \bar{R}_u)}{\sqrt{\sum_{u \in U}(R_{u,i} - \bar{R}_u)^2}\sqrt{\sum_{u \in U}(R_{u,j} - \bar{R}_u)^2}}$$

Once made a model using one of the similarity measures described a, we can predict the rating for any user-item pair by using the idea of weighted sum. First we take all the items similar to our target, and from those similar items, we pick items which the active user rated. By weight the user's rating for every of these items by the similarity between the target item. Finally, we scale the prediction by the sum of similarities to get a value for the predicted rating:

$$P_{u,i} = \frac{\sum_{\text{all similar items, N}}(s_{i,N} * R_{u,N})}{\sum_{\text{all similar items, N}}(|s_{i,N}|)}$$

B. *Algorithm 2: Natural Language Processing*

We convert unstructured data to structured data using following method in NLP
Tokenization
Stop Word Removal
Stemming

C. *Algorithm 3: Support Vector Machine*

Using SVM algorithm, we classify the post, comments, tweets into following two categories
1) Positive
2) Negative

Compare the User based CF and Item based CF result and SVM result and recommend the places as per the search query
There are 2 kinds of SVM classifiers:
Linear SVM Classifier
Non-Linear SVM Classifier

1) *Svm Linear Classifier:*

In the linear classifier model, we assumed that train in examples plotted in space. These data points are expected to be separated by an apparent. It predicts a straight hyperplane dividing 2 classes. The primary focus while drawing the hyperplane is on maximizing the distance from hyperplane to the nearest data point of either class. The drawn hyperplane called as a maximum-margin hyperplane

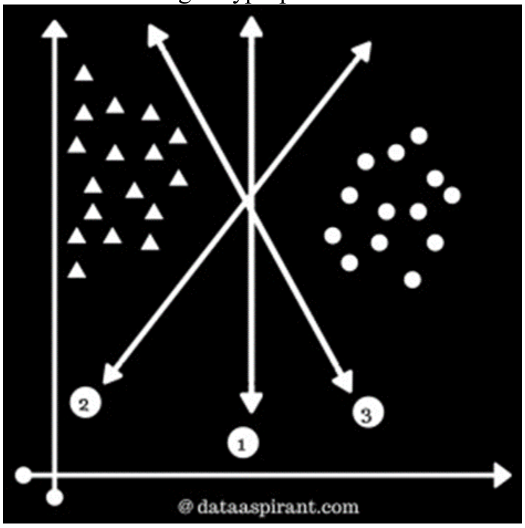
2) *SVM Non-Linear Classifier:*

In the real world, our dataset is generally dispersed up to some extent. To solve this problem separation of data into different classes on the basis of a straight linear hyperplane can't be considered a good choice. For this Vapnik suggested creating Non-Linear Classifiers by applying the kernel trick to maximum-margin hyperplanes. In Non-Linear SVM Classification, data points plotted in a higher dimensional space.

3) *Examples of SVM boundaries*

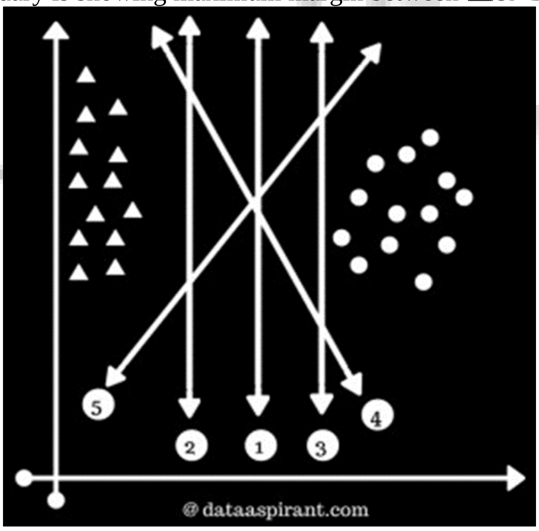
In this section, we will learn about selecting best hyperplane for our classification. We will show data from 2 classes. The classes represented by triangle △and circle○.

Case 1: Consider the case in Fig 2, with data from 2 different classes. Now, we wish to find the best hyperplane which can separate the two classes. Please check Fig 1. on the right to find which hyperplane best suit this use case. In SVM, we try to maximize the distance between hyperplane & nearest data point. This is known as margin. Since 1st decision boundary is maximizing the distance between classes on left and right. So, our maximum margin hyperplane will be "1st "



Case 2:

Consider the case in Fig 2, with data from 2 different classes. Now, we want to find the best hyperplane that can separate the two classes. As data of each class is distributed on left or right. Our motive is to select hyperplane which can separate class with maximum margin .In this case, all decision boundaries are separating classes but only 1st decision boundary is showing maximum margin between △& ○.

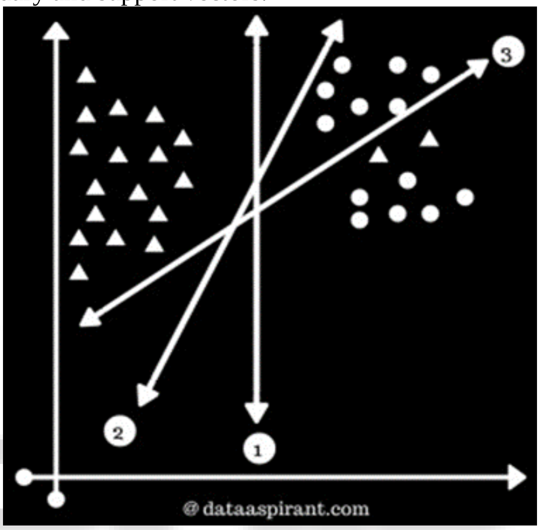


Case 3:

Consider the case in Fig 3, with data from 2 different classes. Now, we wish to find the best hyperplane which can separate the two classes. Data is not evenly distributed on left and right. Some of the △are on right too. You may feel we can ignore the two data points above 3rd hyperplane but that would be incorrect. VM tries to find out maximum margin hyperplane but gives first priority to correct classification1st decision boundary is separating some △ from ○but not all. It's not even showing good margin.2nd decision boundary is separating the data points similar to 1st boundary but here margin between boundary and data points is larger than the previous case.3rd decision boundary is separating all △from all ○classes. So, SVM will select 3rd hyperplane.

Case 4:

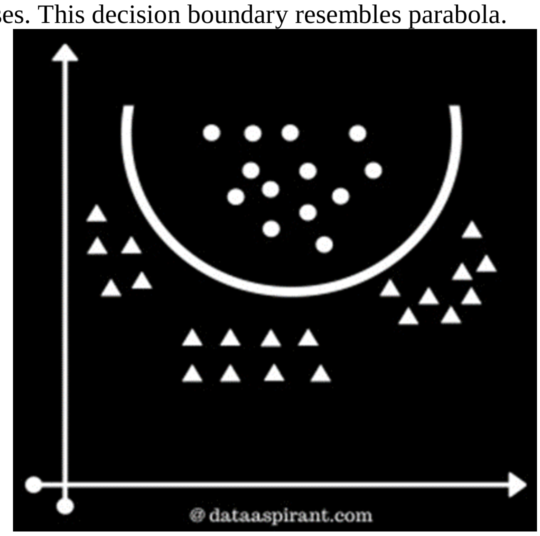
Consider the figure 4, we will learn about outliers in SVM. We wish to find the best hyperplane which can separate the two classes. Data is not evenly distributed on left and right. Some of the △are on right too. In the real world, you may find few values that correspond to extreme cases i.e, exceptions. These exceptions are known as Outliers. SVM have the capability to detect and ignore outliers. In the image, 2 △'s are in between the group of ○. These △'s are outliers. While selecting hyperplane, SVM will automatically ignore these △'s and select best-performing hyperplane.1st & 2nd decision boundaries are separating classes but 1st decision boundary shows maximum margin in between boundary and support vectors.



Case 5:

We will learn about non-linear classifiers. Please check the figure 5 on right. It's showing that data can't be separated by any straight line, i.e, data is not linearly separable. SVM possess the option of using Non-Linear classifier. We can use different type of kernels like Radial Basis Function Kernel, Polynomial kernel etc.

We have shown a decision boundary separating both the classes. This decision boundary resembles parabola.



Linear Support Vector Machine Classifier In Linear Classifier, A data point considered as ap-dimensional vector(list of p-numbers) and we separate points sing (p-1) dimensional hyperplane. There can be many hyperplanes separating data in a linear order, but the best hyperplane is considered to be the one which maximizes the margin i.e., the distance between hyperplane and closest data point of each class. The Maximum-margin hyperplane is determined by the data points that lie nearest to it. Since we have to maximize the distance between hyperplane and the data points. These data points which influences our hyperplane are known as support vectors.

Non-Linear Support Vector Machine Classifier Vapnik proposed Non-Linear Classifiers in 1992. It often happens that our data points are not linearly separable in a p-dimensional (finite) space. To solve this, it was proposed to map p-dimensiona l space into a much higher dimensional space. We can draw customized/non-linear hyperplanes using Kernel trick. Every kernel holds a non-linear kernel function. This function helps to build a high dimensional feature space. There are many kernels that have been developed.

Some standard kernels are:Polynomial (homogeneous) Kernel: The polynomial kernel function can be represented by the above expression. Where k(xi, xj) is a kernel function, xi & xj are vectors of feature space and d is the degree of polynomial function. Polynomial(non-homogeneous) Kernel: In the non-homogeneous kernel, a constant term is also added. The constant term "c" is also known as a free parameter. It influences the combination of features. x & y are vectors of feature space. Radial Basis Function Kernel It is also known as RBF kernel. It is one of the most popularkernels. For distance metric squared euclidean distance is used here. It is used to draw completely non-linear hyperplanes where x & x' are vectors of feature space. is a free parameter. Selection of parameters is a critical choice. Using a typical value of the parameter can lead to overfitting our data. Linearly Separable: For the data which can be separated linearly, we select two parallel hyperplanes that separate the two classes of data, so that distance between both the lines is maximum. The region b/w these two hyperplanes is known as "margin" & maximum margin hyperplane is the one that lies in the middle of them.

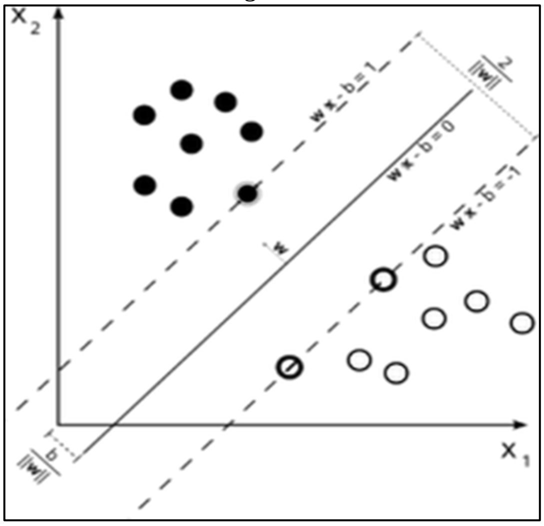
$$\vec{w}x_i - b \geq 1 \text{ if } \theta_i = 1$$
$$\vec{w}x_i - b \leq 1 \text{ if } \theta_i = -1$$

where is normal vector to the hyperplane, θi denotes classes & xi denotes features. The Distance between two hyperplanes is , to maximize this distance denominator value should be minimized i.e, should be minimized.

For proper classification, we can build a combined equation:

$$\|\vec{w}\|_{min} \text{ for } \theta_i \left(\vec{w}x_i - b\right) \geq 1 \ \forall \mathrm{i} = 1, 2, \ldots, \mathrm{n}$$

Non-Linearly Separable: To build classifier for non-linear data, we try to minimizeHere, max() method will be zero(0), if xi is on the correct side of the margin. For data that is on opposite side of the margin, the function's value is proportional to the distance from the margin. where, determines tradeoff b/w increasing the margin size and that is on correct side of the margin.



VI. SYSTEM ARCHITECTURE

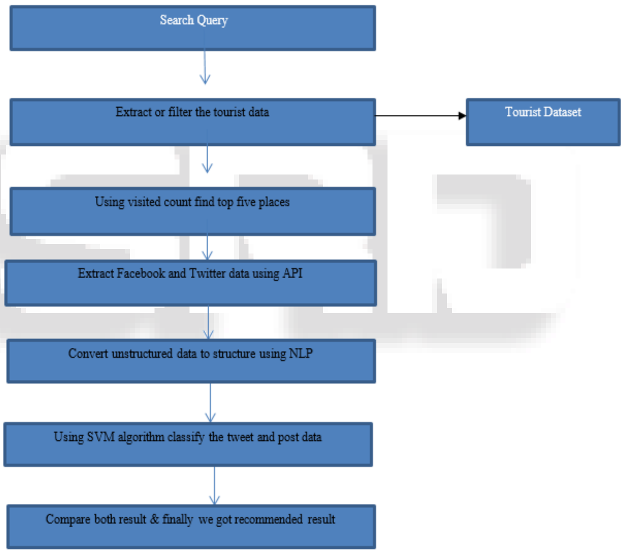
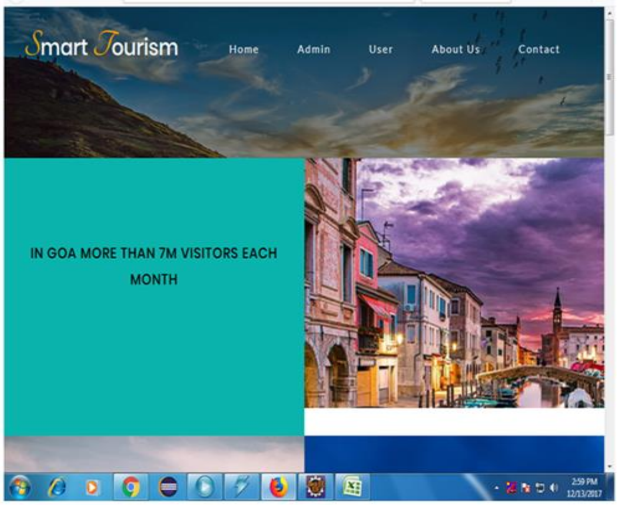
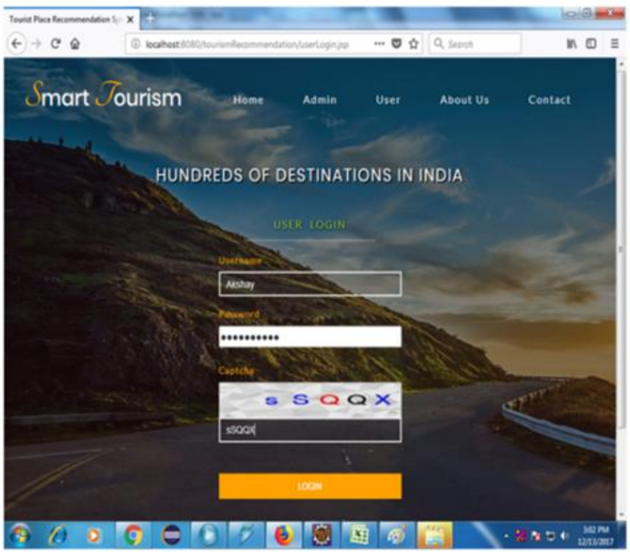
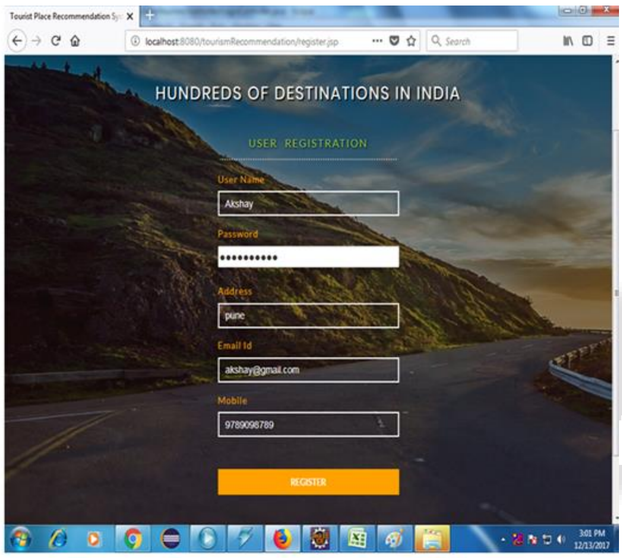


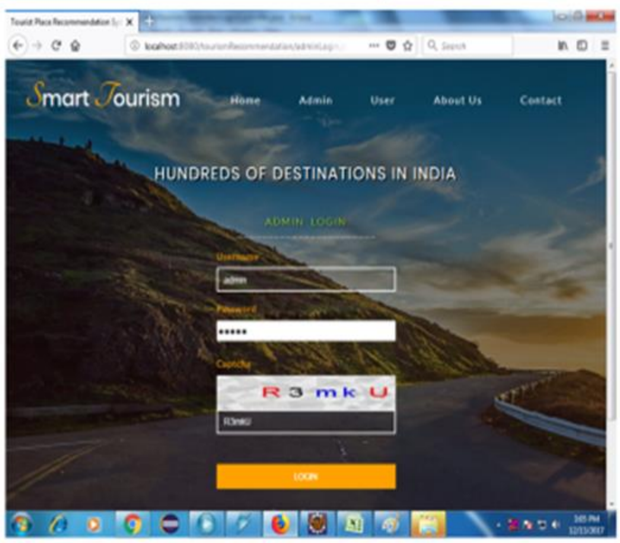
Fig. 1: System Overview

A. Desired Output









VII. CONCLUSIONS

This final section contains a brief summary of the work presented in this paper, describing some points that should be taken into account by scientists aiming to design and develop tourism recommender systems, and an outline of several lines of future work.

VIII. LINE OF FUTURE WORK

This section comments three relevant issues that are currently being studied for the development of e-Tourism recommenders: the diversification of the suggestions provided to the user, the use of social data available in current Web 2.0 applications, and the improvement of the recommendations by the extra capabilities of mobile devices

REFERENCES

[1] Ibrahim Mashal, Osama Alsaryrah, and Tein-Yaw Chung, “Analysis of Recommendation Algorithms for IoT”, Depat of Computer Science and Engineering,Yuan Ze University, Taoyuan, Taiwan, MA, IEEE, 2016
[2] Yassmeen Salman, Abdallatif Abu-Issa, Iyad Tumar, and Yousef Hassouneh, “Proactive Multi-Type Con-text Aware : Recommender System in the Environment of IoT”, Faculty of Engineering and Technology Birzeit University, Ramallah, Palestine, IEEE, 2015
[3] Shunui Jiang, Xueming Qian,Jialie Shen,Yun Fu,Tao Mei, “Author Topic Based Collaborative Filtering For Personalized POI”, Cambridge, MA, IEEE, June 2015
[4] Mohammed F. Alhamid, Majdi Rawashdeh, Haiwei Dong, M Anwar Hossain, and Abdulmotaleb El Saddik, “Exploring the latent preferences for context aware personalize Recommendation systems”, King Saud University, Riyadh, IEEE, Aug2016
[5] Husnul Khotimah,Tua_k Djatna, and Yani Nurhadryani, “Tourism Recommendation Based on Vector space model”, Graduate School of Computer Science,Bogor Agricultural University, ICACSIS, 2016
[6] Yanchang Zhao, Yonghua Cen, “Challenges for Quality of data in smart cities”, Cambridge, MA, Elsevier, Dec2013
[7] Bartlomiej Twardowski, Dominik Ryzko, “IoT and Context-aware mobile recommendations using multi agent systems”, Warsaw University of Technology, Allegro Group, 665{672, IEEE, 2015
[8] In-young Ko, Han-gyu Ko,Angel Jimenez Molina,Jung-hyun Kwon, “Toward a usercentric IoT Based Service Framework”, Korea Advanced Institute of science and Technology, Korea, IEEE, Apr2016
[9] Yunchuan sun, Houbing song,Antonio J. Jara,Rongfang Bie , “Internet of Things and Big Data Analytics for smart and connected communities”, National Scienc of China, IEEE, Apr20 16
[10] Haifeng Liu, Xiangjie kong, Xiaomei Bai, Wei Wang, Teshome Megersa Bekele and Feng Xia, “Context Based Collaborative Filtering for Citation Recommendation”, School of Software, Dalian University of Technology Korea Ad, Dalian, China, IEEE, Mar2015