A

dissertation

on

A TOURISM RECOMMENDER SYSTEM USING GEOTAGGED PHOTOS AND WEATHER FORECASTING

Submitted in partial fulfilment of the requirements for the award of degree of

Master of Technology
in
Computer Science and Engineering

Submitted by

Himanshu Sagar
(16535016)

Under the guidance of

Dr. Manoj Mishra
Professor, Dept. of Computer Science and Engineering

INDAIN INSTITUTE OF TECHNOLOGY, ROORKEE

Department of Computer Science

May, 2018

AUTHOR’S DECLARATION

I declare that the work presented in this dissertation with title "A tourism recommender system using Geotagged Photos and weather forecasting" towards fulfillment of the requirement for the award of the degree of Master of Technology in Computer Science & Engineering submitted in the Department of Computer Science & Engineering, Indian Institute of Technology, Roorkee, India is an authentic record of my own work carried out during the period of May 2017 to May 2018 under the supervision of Dr. Manoj Misra, Professor, Department of Computer Science and Engineering, Indian Institute of Technology Roorkee, Roorkee, India. The content of this dissertation has not been submitted by me for the award of any other degree of this or any other institute.

Date: .... HIMANSHU SAGAR
Place: ROORKEE (16535016)
M.TECH (CSE)

CERTIFICATE

This is to certify that the statement made by the candidate is correct to the best of my knowledge and belief.

Date: .... Sign: ....
Place: .... Dr. Manoj Misra
(Professor)
IIT Roorkee

ACKNOWLEDGEMENTS

Dedicated to my family and friends, for standing by me through thick and thin, without whom I would not have gotten this far. I would like to express my sincere gratitude to my advisor Dr. Manoj Misra for the continuous support of my study and research, for his patience, motivation, enthusiasm and immense knowledge. His guidance helped me in all time of research and writing of this thesis. I could not have imagined having a better advisor and mentor for my study.

I am also grateful to the Department of Computer Science and Engineering, IIT Roorkee for providing valuable resources to aid my research.

HIMANSHU SAGAR

Abstract

Whenever an individual start planning for a trip, one basic question arises in front of him, “where”? This problem of finding a tourist place is very common in today’s world. At first the suggestion for place to visit, comes from the family member and friends. The knowledge of individual and his friends might be limited as they can suggest only those place that they have visited personally. Another option comes to the mind of user to visit travel agency. This option is also flawed since travel agent might be biased and only suggest places which are beneficial to him. The final option user have to log on the World Wide Web and check for yourself about the tourist destination. When users choose the third option to make a decision, generally it makes it even harder to choose a place to visit. World Wide Web have unlimited amount of data and when this data is thrown at the user, who lack the experience in dealing with situation like this, he gets confused.

This large amount of data and lack of experience in user creates a demand for a system which can help user to access this data and provide better result. These result can help user with the decision making process. In this report, a method is proposed for such a system. This system access huge amount of image data and provide results on the basis of the user query. Those results can help user in decision making process for the tourist destination.

Table of Contents

List of Figure and Tables

1. Introduction

Many people depend on the World Wide Web to plan their tourism trips. They usually go to the multiple websites which contains the information regarding various tourist destination and which time is best for them to go at that place. However, World Wide Web have enormous amount of data. Usually that amount of data which is supplied to end consumer did not end up helping him instead it creates extreme confusion in the mind of the consumer about the trip planning. People also plan their trips on the suggestion of their family members and friends. This way for planning a trip also have limitation. The knowledge about the tourist destination is very limited among the friends and family members. Only those places will be suggested to you which have been visited by them. Another way to plan a trip is to contact a travel agency. They might be having much better knowledge about tourist destination than end user's family and friends but they can be heavily biased. They tend to suggest services or business which have tie up with that particular agency.

While it is important to choose an interesting location one question also comes to the mind into the end user which is "when to visit". Dynamic factors such as weather also plays an important role in choosing of the tourist destination. There are a no. of platforms available which can provide weather forecasting services such as OpenWeatherMap and AccuWeather. It'll be a really tiresome approach for an individual to choose a location out of hundreds of location that have been suggested to him by websites/people and also checking for the weather on the time period of visiting that location.

Tourists all over the world tend to take picture of landmarks and interesting places while travelling. These pictures are also shared on the internet on platform Flickr, Facebook, Instagram etc. These platforms allows them to share these images along with the informative tags and metadata i.e. geotags. Most of the photo capturing devices that tourist uses comes with GPS receivers i.e. these devices also capture the location in term of GPS coordinates. Geotags denotes the location of the captured pictures. Flickr is one of the most popular website to share pictures. Flickr have over 90 million user [1]. Average no. of pictures uploaded in a day is 1 million [1]. There are over 10 billion images available on this platform which makes it really data enrich platform. This amount of data can also cause problem. There could be a lot of geotagged images shared on the platform which doesn’t indicate any tourist location. This amount of data should be handled and preprocessed carefully.

Facing these situations end user is in requirement of a system which can recommend him tourist destination in an efficient and time-saving manner. So that he doesn’t have to look up into hundreds of website and will be able to choose a tourist destination without any third party interference.

1.1 Motivation

Planning of a trip is one of the key element for the success of overall trip. People plan their trip according to their preferences. Sometimes people doesn’t have enough information to make the decision. Even if they rely on third party for help in the decision making process, there is chance of lack of information or biased opinions might influence the end result.

This lack of information can be tackled using the one of the most relevant technology to the user, Internet. Internet can provide a large no. of option along with the data to support each option. This overwhelming amount of data might help some users but most of users gets confused because they do not have the skill to access all that data in a meaningful manner. Tools are in requirement for these user to tackle these situation in more professional manner.

The overwhelming amount of information, tools and inexperience of user to deal with these kind of situation makes it clear that some kind of system should be presented to help in the decision making process. These automatic/semiautomatic system should be able to access that data and provide results on the basis of user queries which can solve a challenging problem for the user.

1.2 Objective

The objective of this thesis start from the presupposition that development of tourism recommender system can help user in the decision making process of choosing a tourist destination. In this report we propose a development of recommender system using geotagged photos and weather data.

- The recommender should be able to provide results on the basis of the user queries.
- This system should be able to handle huge amount of image data in order to process them.

- System should also consider the weather while recommending the places to visit.

1.3 Problem Statement

To develop a tourist recommendation system to help the user in choosing a tourist destination. Recommendation system should be able to provide results on the basis of user query. It should also consider the effect of the dynamic nature of weather while finding results.

1.4 Organization of Thesis

This dissertation is organized as follows:

- Chapter 1: Introduction about the topic. Motivation, objective and problem statement
- Chapter 2: Related works. Discusses the work done in the field of the recommender system and geo data.
- Chapter 3: Proposed Approach. Problem description and discussion of system that is defined considering the objective of this dissertation.
- Chapter 4: Experiment and results. Contains the results of the proposed system
- Chapter 5: Conclusion and Future Works. It concludes the dissertation and listing the areas to be worked on.

2. Related Works

2.1 Recommender System

There are multiple search system which allows to user choose out of multiple choice of his preference. Yahoo Travel [2] and Google Maps [3] are the prime example of such system. User can search places on the basis of timing and user ratings on such system. Using GPS trajectories Takeuchi et al [5] proposed a shop recommending system. In that work authors applied location data to the collaborative filtering algorithm, this algorithm had been utilized in many online recommendation systems, by transforming location data history into a list that contains the names of each user's frequently visited shops and rating values which indicate how fond the user is of each shop. In a System proposed by Cheng et al. [6] personalized attributes are taken into consideration for making the recommendation of the users. They downloaded the photos from the online media sharing website Flickr [4]. Then Using Mean-shift based method on the pictures to generate point of interest in each city. Now the face detection is applied on the downloaded photos. Another attributes are generated using people attribute detector. Travel path are mined using these photos and people attribute for example if majority of face detected as a male in a particular place then it is supposed that the particular place is favored by male population. Zheng et al. [7] proposed a travel recommendation system on the basis of GPS history data. They extracted Point of interest in a city. They also performed activity-activity correlation matrix. Having the data of location-activity matrix, location-feature matrix and activity-activity correlation matrix, they trained a recommender system. Zheng and Xie [8] also proposed a personalized approach for the recommender system. They first modeled different users' location history with their framework. For personalization module they calculated the correlation between locations using end user's travel experience and sequence of the visited location. Using this data they predicted user's interest in unvisited location. In paper by Cao et al [11] they proposed an approach for the recommender system. They created dataset from the online platform Flickr [4]. Then they used Mean-shift clustering algorithm for clustering of those images. And in results using query image from the user on the basis of similarity of other pictures recommendation is provided to the user.

Data mining from the GPS trajectories is also related to the recommender task because the ultimate goal is to predict where a person may be going. There are various method proposed for the prediction of next location using mining. A detailed review of the methods of mining for trajectory is given by JD Mazimpaka et al. [9]. Other previous works are focused on the analysis of historical trajectories, concluding that human mobility behavior have high degree of the regularity. Patterns and models can be extracted from the set of trajectories. Wesley Mathew et al [10] have classified these modelling approaches into 3 distinct category. (a) State Space models, (b) Data mining technique and (c) Template matching Techniques. In state space models the variation in the spatial sequences is captured through sequence models such Markov Model and its extensions. These models generally have higher training complexity due to dealing with uncertainty. In case of location prediction Markov models can naturally be used because they support the series of possible future visits. In data mining techniques we try out frequent patterns and association rules. Trajectory is defined as an ordered sequence. Most of the data mining method tries to maximize confidence on the basis of previous occurrences and do not consider any spatial and temporal element in the analysis of the patterns. In template matching, the extracted features are compared to pre-stored patterns, using similarity metrics. These approaches also suffers with high time run complexity.

Type of recommender system.

In [15] by Borràs et al. authors have defined the type of recommender system according to the approaches that have been used. Fig 2.1 shows the classification of recommender System.

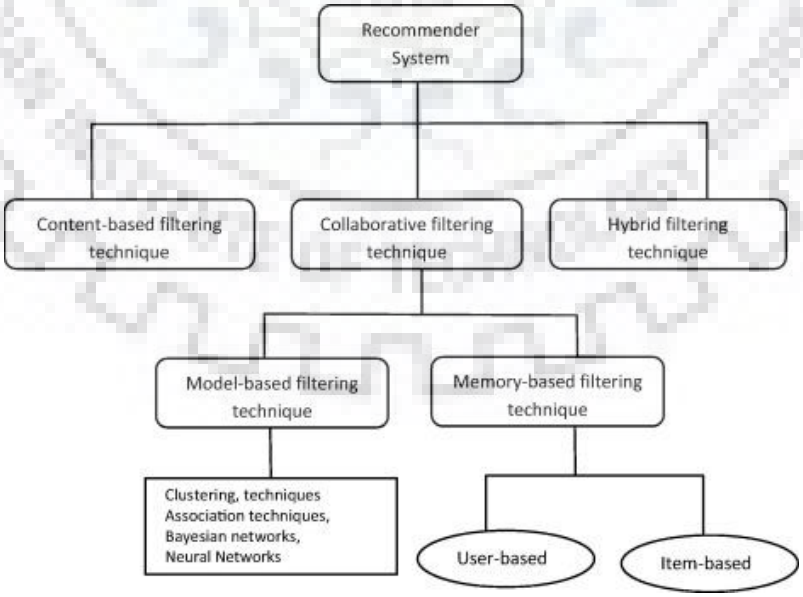

Recommender System
Content-based filtering technique
Collaborative filtering technique
Hybrid filtering technique
Model-based filtering technique
Memory-based filtering technique
Clustering, techniques
Association techniques,
Bayesian networks,
Neural Networks
User-based
Item-based

Figure 2.1 Classification of recommender System.

In Content based filtering technique is domain dependent technique i.e. this technique calculate a degree of similarity between the users and the items to be recommended. This type of approach may suffer from the “cold start” problem. If the database is not metadata enriched then the results using this approach might not be accurate.

In Collaborative filtering technique is used when the metadata is not in adequate amount. It make the recommendation on the basis of the users with similar preference. This type of approach suffers from “grey sheep” problem. This problem arises in the early stage. If a user doesn’t have matching preference with other users, results will not be accurate.

Hybrid approach is to develop the recommender system using both of the technique described above.

2.2 Geotags exploration

Flickr have a huge database and it contains over 10 billion photos. Anyone can access that database using FlickrAPI, using which user can search or download flickr databases for images. A lot of approaches have been proposed to map these geotags in geographical region. Metadata such as tags, title and time can also be used to symbolize these geographical region. In [7] by Zheng et al. they used GPS traces to define a method to extract interesting locations. Cao et al [11] uses geo tag to map the location in geographical region. They use mean-shift clustering algorithm to for clustering of all those images.

In this work we also follow the mainstream approach which is being followed by most authors. The images data will be gathered from the FlickrApi and mean-shift clustering algorithm will be used to cluster those images

2.3 Weather Data

There are multiple weather forecasting platform available. Accuweather [12] and OpenWeatherMap [13] are most used platform on the web. These platform provide API to access the services.

In our system we’ll be using OpenWeatherMap. Because it is easy to access and its 13 days forecasting service is beneficial for the system to recommend better places to user.

3. Proposed approach

3.1 Problem description

As we've discussed in chapter 1 the problem arises in the initial part of the planning a trip. People generally get confused because of lack of data or overwhelming amount of data. On the basis of this huge data user is supposed to make a choice but sometime lack of experience causes problem. So the combination of huge amount of data and lack of experience in user creates an urgent need of tool which can deal with that amount of data and provide result to the user. There are tons of images available of the tourist destination along with their metadata. Using those images development of such system is required to help the end user in its decision making process.

The consideration of Dynamic factors should also be included in such system. It improves the results for the End user. If a location is recommended without considering these factors quality of results will be degraded. For ex. a location is suggested to the user but the temperature of that place is way below the average temperature of that city. End user will not be able to enjoy the trip at that location in such condition. Hence the results from system is not satisfactory to user. In order to increase the user satisfaction from the system such modules to consider dynamic factor.

3.2 Proposed Approach

In this section, method is proposed for the recommendation system. This approach is based on the fact that the data uploaded on the online photo sharing website such as Flickr can be used to find landmarks and can be used in recommender system. Huge Data/Images collected from Flickr may contain irrelevant image data. These images may contains the content which is not related to geographical coordinates.

To solve these difficulties we'll use approaches provided by Cao et al. [11]. After downloading the images from the online platform that dataset will be organized and representative sample will be extracted from dataset for future use. They have also given the efficient clustering algorithm to divide the geographical regions on the basis of geographical coordinates and the distribution of geotagged images. After recognizing those cluster, representative images are determined to denote cluster. User can enter a query Image now on the comparison with representative images system will recommend user location. Images shown in results also have the geo tags i.e. it contains the geo graphical location. This data contains latitude and longitude. Using this current or future weather forecasting can be done using OpenWeatherMap. This will help user in the decision making. Figure 3.1 shows the framework of the recommendation System.

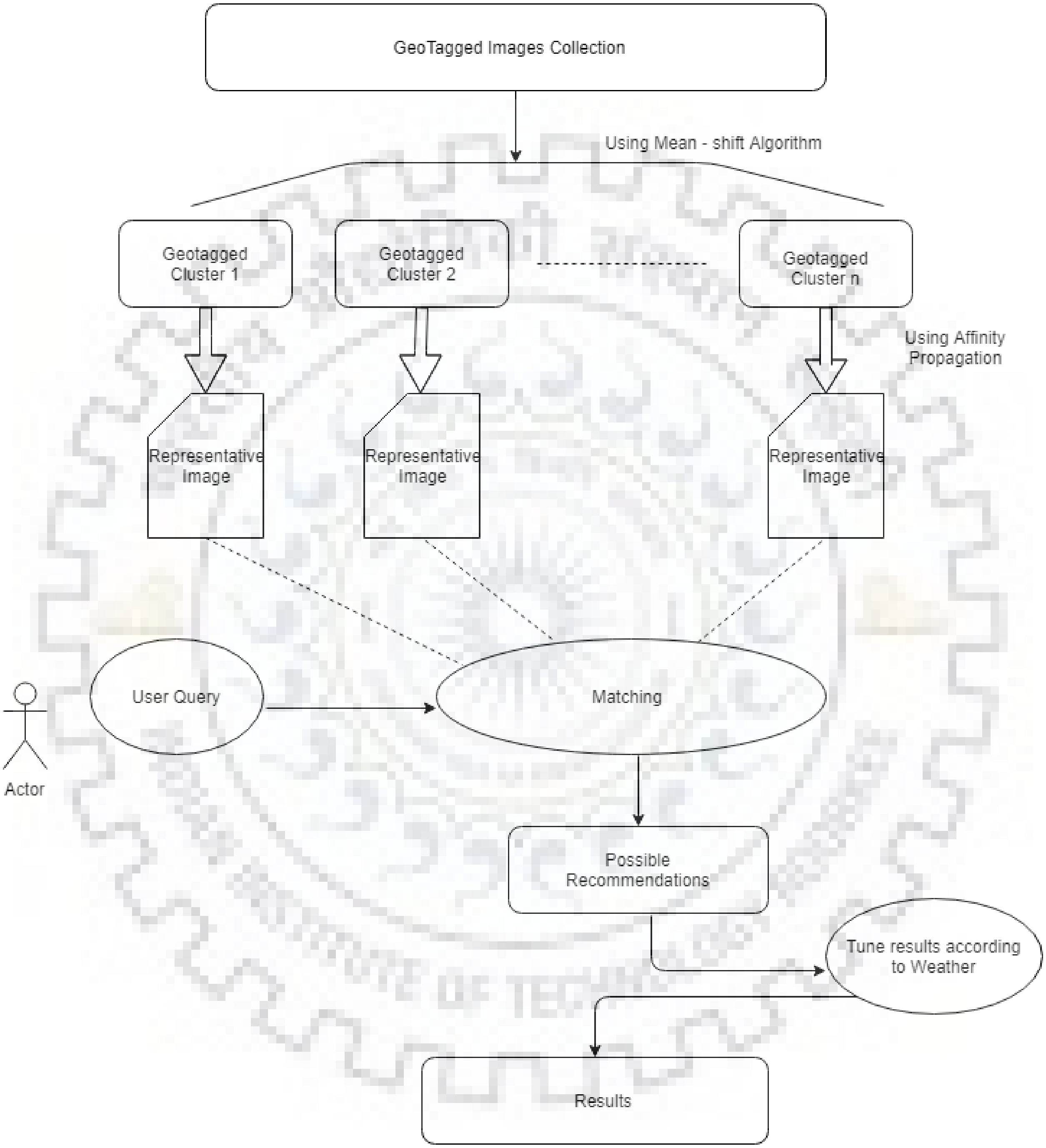


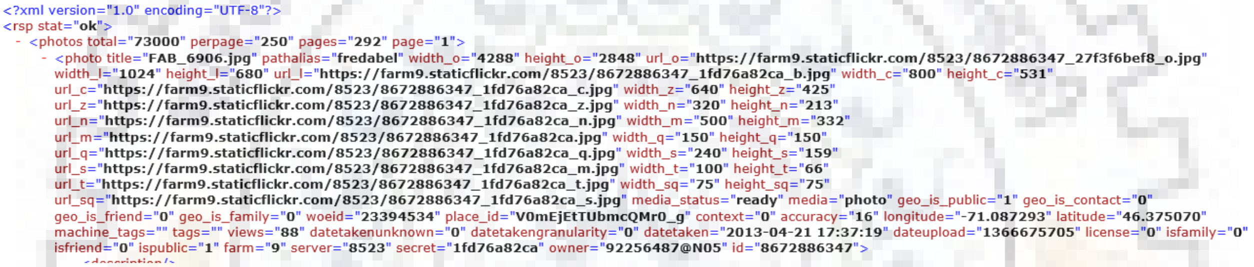
Figure 3.1 framework of the recommendation System

3.3 Dataset

About 10,000 images were downloaded along with their GPS and other metadata from Flickr. FlickrApi Photo search was used to download these images. Initially the data downloaded from this source is an XML form. These XML files contain the data about images in form of tags. The format of data for an image in XML file is shown in figure 3.2.

Assumption in dataset:

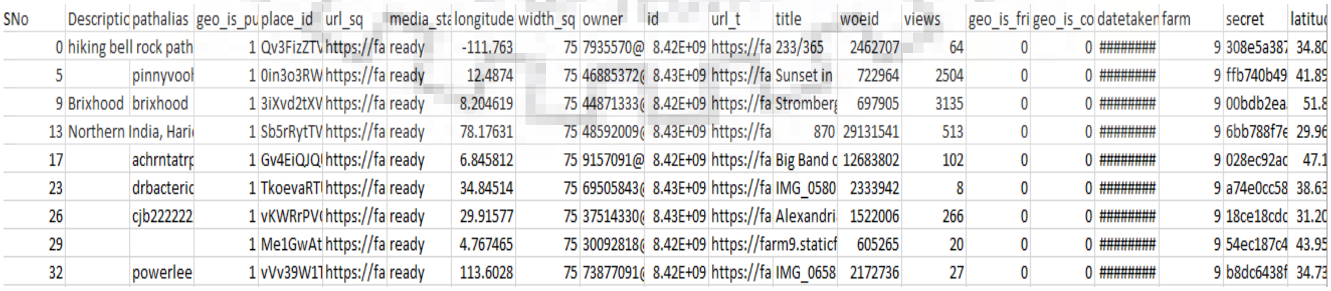
In this dataset we are assuming that photos that were uploaded on Flickr taken by user contains the scenic perspective of the tourist location. For e.g. if tourism location contains Museum, only photos taken from the outside will be used to determine that tourist location. Photos taken inside the museum will not be considered to locate that destination.



```
<?xml version="1.0" encoding="UTF-8"?>
<rsp stat="ok">
 - <photos total="73000" perpage="250" pages="292" page="1">
   - <photo title="FAB_6906.jpg" pathalias="fredabel" width_o="4288" height_o="2848" url_o="https://farm9.staticflickr.com/8523/8672886347_27f3f6bef8_o.jpg"
     width_l="1024" height_l="680" url_l="https://farm9.staticflickr.com/8523/8672886347_1fd76a82ca_b.jpg" width_c="800" height_c="531"
     url_c="https://farm9.staticflickr.com/8523/8672886347_1fd76a82ca_c.jpg" width_z="640" height_z="425"
     url_z="https://farm9.staticflickr.com/8523/8672886347_1fd76a82ca_z.jpg" width_n="320" height_n="213"
     url_n="https://farm9.staticflickr.com/8523/8672886347_1fd76a82ca_n.jpg" width_m="500" height_m="332"
     url_m="https://farm9.staticflickr.com/8523/8672886347_1fd76a82ca.jpg" width_q="150" height_q="150"
     url_q="https://farm9.staticflickr.com/8523/8672886347_1fd76a82ca_q.jpg" width_s="240" height_s="159"
     url_s="https://farm9.staticflickr.com/8523/8672886347_1fd76a82ca_m.jpg" width_t="100" height_t="66"
     url_t="https://farm9.staticflickr.com/8523/8672886347_1fd76a82ca_t.jpg" width_sq="75" height_sq="75"
     url_sq="https://farm9.staticflickr.com/8523/8672886347_1fd76a82ca_s.jpg" media_status="ready" media="photo" geo_is_public="1" geo_is_contact="0"
     geo_is_friend="0" geo_is_family="0" woeid="23394534" place_id="V0mEjEtTUbmcQMr0_g" context="0" accuracy="16" longitude="-71.087293" latitude="46.375070"
     machine_tags="" tags="" views="88" datetakenunknown="0" datetakengranularity="0" datetaken="2013-04-21 17:37:19" dateupload="1366675705" license="0" isfamily="0"
     isfriend="0" ispublic="1" farm="9" server="8523" secret="1fd76a82ca" owner="92256487@N05" id="8672886347">
       <description/>
```

Figure 3.2 XML Data

To access this information in an easy manner the data is required to be in a format which is easily understandable to the system/code. There was total 629 XML file containing data of over 10,000 photos. All of these XML files converted into CSV format. Which contains all of the data in more organized manner. But this data may also contain the duplicate images from tourist. After cleaning duplicate photos we’ll get our dataset in much cleaner format. Example of the dataset is shown in Figure 3.3.



SNo	Descriptic	pathalias	geo_is_pu	place_id	url_sq	media_sta	longitude	width_sq	owner	id	url_t	title	woeid	views	geo_is_fri	geo_is_co	datetaken	farm	secret	latitud
0	hiking bell	rock path	1	Qv3FizZTV	https://fa	ready	-111.763	75	7935570@	8.42E+09	https://fa	233/365	2462707	64	0	0	########	9	308e5a387	34.80
5		pinnyvool	1	0in3o3RW	https://fa	ready	12.4874	75	46885372(	8.43E+09	https://fa	Sunset in	722964	2504	0	0	########	9	ffb740b49	41.89
9	Brixhood	brixhood	1	3iXvd2tXV	https://fa	ready	8.204619	75	44871333(	8.42E+09	https://fa	Stromberg	697905	3135	0	0	########	9	00bdb2ea.	51.8
13	Northern India, Hari		1	Sb5rRytTV	https://fa	ready	78.17631	75	48592009(	8.43E+09	https://fa	870	29131541	513	0	0	########	9	6bb788f7e	29.96
17		achrntatrp	1	Gv4EiQJQ	https://fa	ready	6.845812	75	9157091@	8.42E+09	https://fa	Big Band c	12683802	102	0	0	########	9	028ec92ac	47.1
23		drbacteric	1	TkoevaRT	https://fa	ready	34.84514	75	69505843(	8.43E+09	https://fa	IMG_0580	2333942	8	0	0	########	9	a74e0cc58	38.63
26		cjb222222	1	vKWRrPV(	https://fa	ready	29.91577	75	37514330(	8.43E+09	https://fa	Alexandri.	1522006	266	0	0	########	9	18ce18cdc	31.20
29			1	Me1GwAt	https://fa	ready	4.767465	75	30092818(	8.42E+09	https://farm9.staticf		605265	20	0	0	########	9	54ec187c4	43.95
32		powerlee	1	vVv39W1T	https://fa	ready	113.6028	75	73877091(	8.42E+09	https://fa	IMG_0658	2172736	27	0	0	########	9	b8dc6438f	34.73

Figure 3.3 CSV data

3.4 Mean-Shift Algorithm

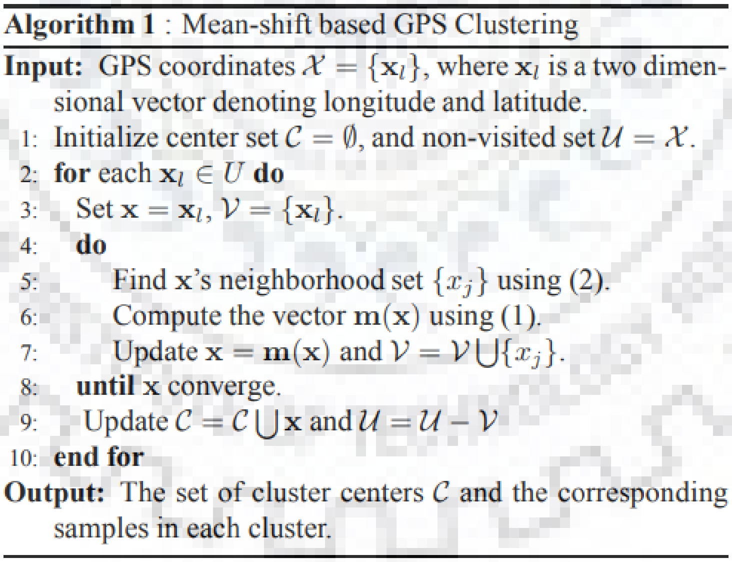
For clustering of these images, Mean-Shift Algorithm [16] is used. Mean shift algorithm is very effective in the area of the image clustering. It is a non parametric method for image clustering and it doesn’t require to specify the number of clusters. It starts from a given Sample x, and looks for the vector

$$m(x) = \frac{\sum_i x_i g_i}{\sum_i g_i}$$

In this g_i is the local kernel density function in the form of

$$g_i = g(\parallel x - x_i)/h \parallel^2)$$

Where g should be non-negative, non – increasing continuous function. In mean shift algorithm finding the closest neighbor is most expensive operation. In [11] authors have proposed to formulate g as a flat kernel and obtain necessary condition for the nearest neighbor. This modified mean shift algorithm is as follows.



```
Algorithm 1 : Mean-shift based GPS Clustering
Input: GPS coordinates $\mathcal{X} = \{\mathbf{x}_l\}$, where $\mathbf{x}_l$ is a two dimensional vector denoting longitude and latitude.
Initialize center set $\mathcal{C} = \emptyset$, and non-visited set $\mathcal{U} = \mathcal{X}$.
for each $\mathbf{x}_l \in U$ do
    Set $\mathbf{x} = \mathbf{x}_l$, $\mathcal{V} = \{\mathbf{x}_l\}$.
    do
        Find x’s neighborhood set $\{x_j\}$ using (2).
        Compute the vector $\mathbf{m}(\mathbf{x})$ using (1).
        Update $\mathbf{x} = \mathbf{m}(\mathbf{x})$ and $\mathcal{V} = \mathcal{V} \bigcup \{x_j\}$.
    until x converge.
    Update $\mathcal{C} = \mathcal{C} \bigcup \mathbf{x}$ and $\mathcal{U} = \mathcal{U} - \mathcal{V}$
end for
Output: The set of cluster centers $\mathcal{C}$ and the corresponding samples in each cluster.
```

Given Algorithm 1 works very well for the low dimensional data. It can cluster huge amount of image data within minutes on a regular pc.

3.5 Affinity propagation

In our proposed method we find the representative image and tags in the cluster that have been provided by Mean shift algorithm. Representative images and tags are denoted by R-images and R-tags respectively. To find the R tags, we look in the description data field of our dataset and count the no. of tags that have been specified by user, if a tag has occurred more than threshold it is considered to be an R-tag.

It is a difficult task to compute R-images. Affinity propagation algorithm is used for this operation.

There are N images in a cluster and image i and image k have some similarity, this similarity between images is calculated by a Gaussian Function

$$S_{(i,k)} = \exp(-\| f_i - f_k\|^2 / \rho)$$

Where f denotes the image feature i.e. color histogram (in our work). ρ is estimated variance of the given features . Using this algorithm we are looking for exemplar for each image. This algorithm considers all data point as tentative exemplars and exchange messages between data points until it determine good results in a set of exemplars. Using these methodology R-images is found in the cluster. Even we have found the potential representative images in each geotagged cluster some of the images might not be meaningful. These images will be removed using comparison on R-images.

These comparison are done using Chi – Square test[17]. Histogram is generated for the images and then that histogram is normalized for comparison. The given equation is used for the chi – Square test.

$$d(H_1, H_2) = \sum \frac{((H_1(I) - H_2(I))^2}{H_1(I)}$$

Where $H_{1,}$ .H_2 are the histogram of images to compare. $d(H_1, H_2)$ is the metric to express the matching of both histogram.

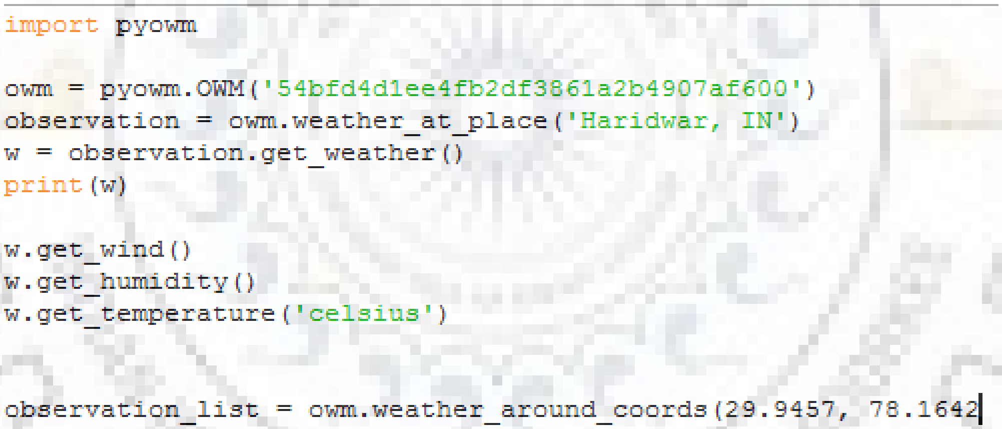
After generating these R-images we begin with our query image. In this approach we prioritize the location closer to the user. This would require location access of the user. We calculate the distance between recommended places and user using haversine formula [18]. In the Haversine formula the distance is given by equation.

$$= 2r \arcsin\left(\sqrt{\sin^2\left(\frac{\varphi_2-\varphi_1}{2}\right)+\cos(\varphi_1)\cos(\varphi_2)\sin^2\left(\frac{\lambda_2-\lambda_1}{2}\right)}\right)$$

Where φ denotes the latitude and ƛ denotes the longitude. Using the given equation distance can be measured and the recommendation can be tuned according to the reachability.

3.6 Weather APIs

After getting the user query and matching it with R-images or R-tags the potential results is generated i.e. set of images. Now these images contains the GPS location of the place where these pictures were taken. Using those GPS coordinate we'll find the current and future weather forecast on those locations. Then on the basis of the weather forecast result will be displayed to the user. GPS coordinates will be entered in the system and using API provide by OpenWeatherMap. From this API we get temp and other factors for e.g. wind, humidity as shown in fig 3.4

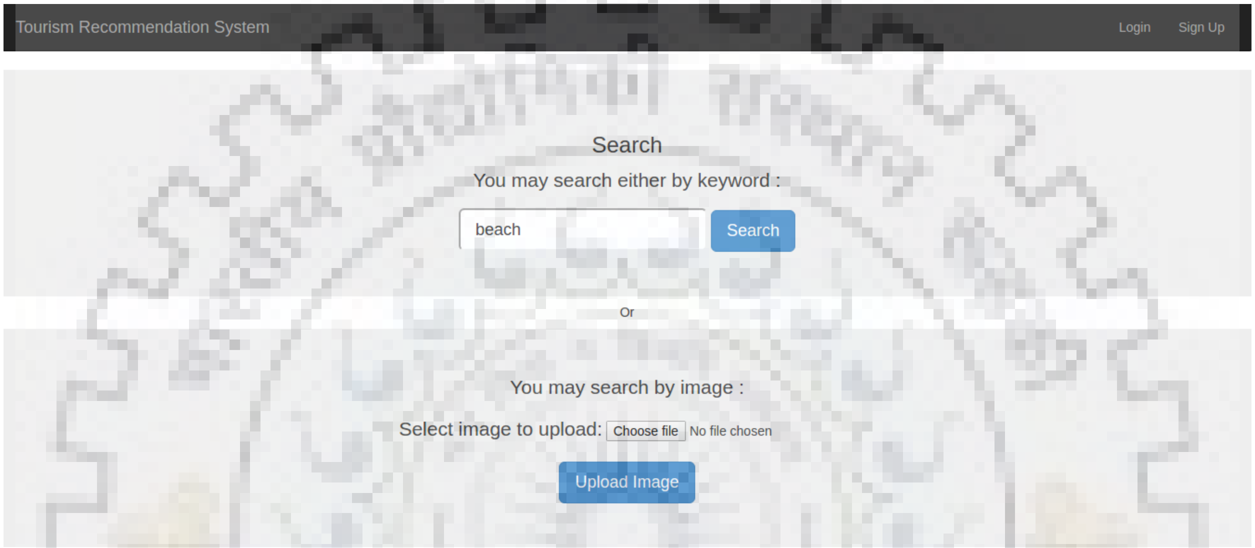


```
import pyowm

owm = pyowm.OWM('54bfd4d1ee4fb2df3861a2b4907af600')
observation = owm.weather_at_place('Haridwar, IN')
w = observation.get_weather()
print(w)

w.get_wind()
w.get_humidity()
w.get_temperature('celsius')


observation_list = owm.weather_around_coords(29.9457, 78.1642
```

Figure 3.4 Weather API

After finalizing the results on the basis of the reachability and the weather some information is also provided to the user. This information can be provided using descriptive tag that are associated with images or using the GPS coordinates we can try to find out about the place. We are choosing latter methodology as descriptive tag might have some non-relevant information to the user. Using the Google Api we are trying to fetch information about the places that have been recommended to the user so far. The information contain the near places to visit around those GPS coordinates. This will help user to make much better decision while choosing place to visit.

The interface for this recommender system will be web based. User have to access the interface the through Internet. User can search through tag and image in that interface. Interface is shown in figure 3.5



Search

You may search either by keyword :

beach

Or

You may search by image :

Select image to upload: No file chosen

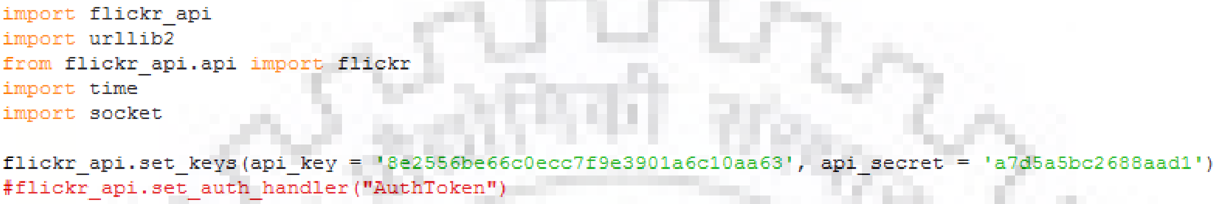
Figure 3.5 User Interface

User can upload image to search them or he can use keyword or tags. Results will be displayed after processing the user query.

4. Experiments and Results

4.1 Experiment Environment

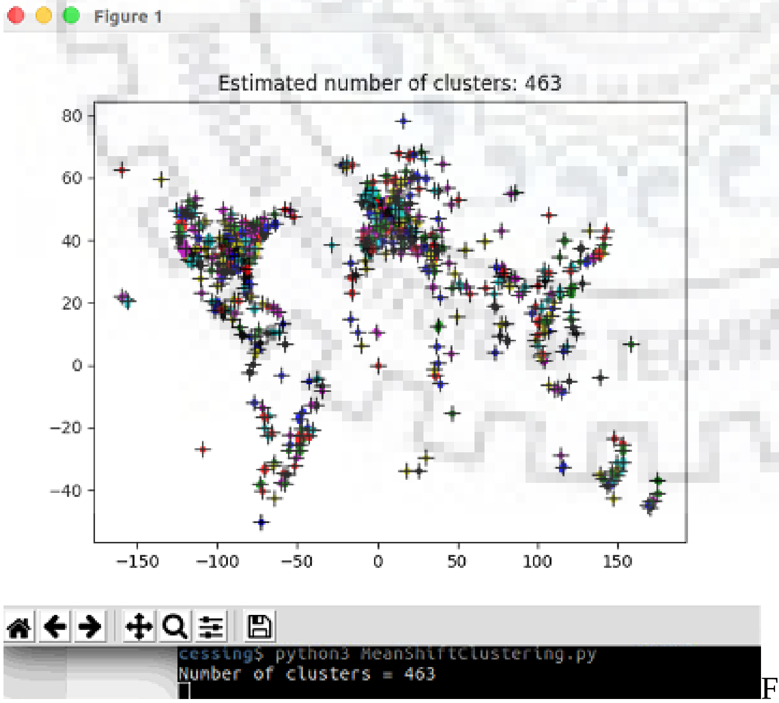
We’ve downloaded the xml files using FlickrApi. The format of these XML files as shown in figure 2. To use the flickr.photos.search api to download administration needs to have authentication key and secret. Our api key and api secret is shown in Figure 4.1



```
import flickr_api
import urllib2
from flickr_api.api import flickr
import time
import socket

flickr_api.set_keys(api_key = '8e2556be66c0ecc7f9e3901a6c10aa63', api_secret = 'a7d5a5bc2688aad1')
#flickr_api.set_auth_handler("AuthToken")
```

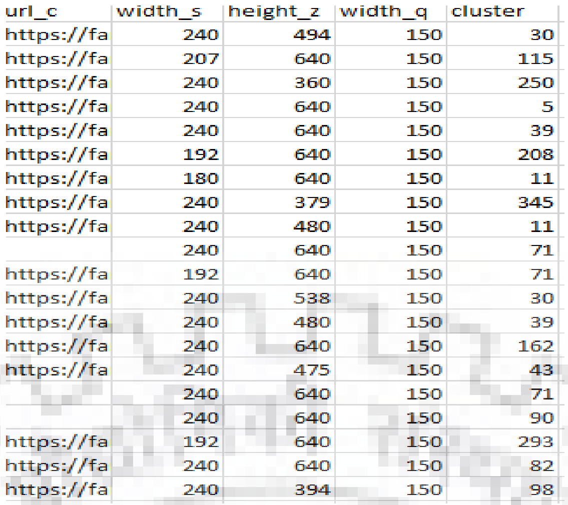
Figure 4.1 Flickr API key and Secret

After storing these xml files, all of them are converted into csv file for better organized dataset. The format of csv dataset is shown in figure 3. After removing the duplicates from the dataset mean shift clustering algorithm which is shown in Algorithm 1 is applied on the preprocessed dataset. It’ll give the no. of clusters on the basis of the latitude and longitude. In our case 463 clusters were formed using those images. Figure 4.2 shows the estimated no. of clusters and figure 4.3 denotes the no. of cluster associated to the picture in the csv data format.



```
cessing$ python3 MeanShiftClustering.py
Number of clusters = 463
```

Figure 4.2 Clusters



url_c	width_s	height_z	width_q	cluster
https://fa	240	494	150	30
https://fa	207	640	150	115
https://fa	240	360	150	250
https://fa	240	640	150	5
https://fa	240	640	150	39
https://fa	192	640	150	208
https://fa	180	640	150	11
https://fa	240	379	150	345
https://fa	240	480	150	11
	240	640	150	71
https://fa	192	640	150	71
https://fa	240	538	150	30
https://fa	240	480	150	39
https://fa	240	640	150	162
https://fa	240	475	150	43
	240	640	150	71
	240	640	150	90
https://fa	192	640	150	293
https://fa	240	640	150	82
https://fa	240	394	150	98

Figure 4.3Cluster among Images

After formation on the cluster on the basis of the two feature space, longitude and latitude next step is to find R-images and R-tags for each cluster.

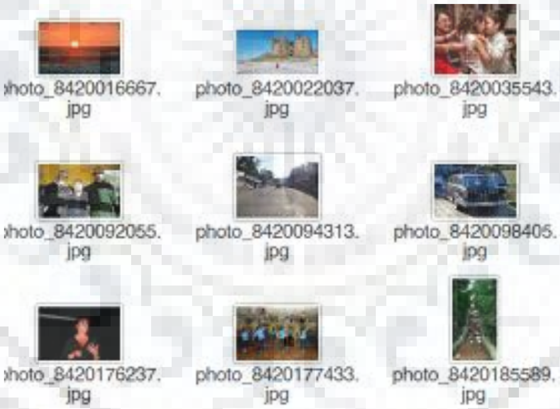


Figure 4.4 Extracted Images

Figure 4.4 shows the images extracted after clustering of the images using mean shift algorithm. After this we need to find R-images and R-tags to represent theses cluster. By performing these operation we have to compare user query just with the represent image.

We have used affinity propagation methodology for finding the R-images. R tags can be found by checking the description or tags of the image. If a tag crosses the threshold in number of count it have appeared which is 6 in our system it is considered to be an R tag. To find the R images for each cluster a similarity feature is used. For each cluster an R image is picked which have similar features for the rest of images of cluster. To find this similarity measure we've used histogram.

We are determining exemplar C_i for each image i, where C_i = 1,2….., N using affinity propagation. Here C_i = i1 which mean image is itself exemplar hence it is a representative image. This algorithm consider all data points as potential exemplars and continuously exchanges messages between them and only halt the process when it find a good solution. There are two types of messages: responsibility $r_{(i,k)}$ which denotes the confidence of image i belonging to cluster k, and other message is availability $a_{(k,i)}$ it represents the possibility of image k being the exemplar of k. This algorithm updates $r_{(i,k)}$ and $a_{(k,i)}$ continuously in an iterative manner until convergence of the image. At last the exemplar for image is selected by

$$p_i = argmax_k[r_{(i,k)} + a_{(k,i)}]$$

Now on the basis of user query we will be able to find pictures which should be recommended to the user. However, before suggesting these images along with their data to the user we have to check dynamic factors too i.e. weather. The potential result is stored. Now from another module which is used to calculate the weather forecasting this potential result is accessed. Using the geo tags of images that are stored in results we are able to see the current and future weather forecasting. There could be multiple way to categories destination according to the weather. We are using much simpler approach for this. We are just looking for extreme situations for ex. rain fall, storm, heavy snowfall and extreme temperatures. If the location have temperature between $18^{o}C$ to $32^{o}C$ they have score of 1. If the location have temperature out of the given bound they have score of 0.75. If they have extreme situation described as above score of 0.5 will be associated to the place. After sorting according to the weather score these recommendation are shown to the user.

4.2 Results

Tags can be easily accessed from the description field of the dataset. Image query is not a trivial task to complete. Affinity propagation module is used to tackle this. Chi-squared method is used to compare the histogram of the query image and representative images. Histogram of both images are generated and normalized and then compared to find out matching parameter. R images which have the most matching parameter to the query images are inserted into results. And after sorting those results according to the weather score that have been decided earlier and display the results.

For results we've used a beach image shown in figure 4.5

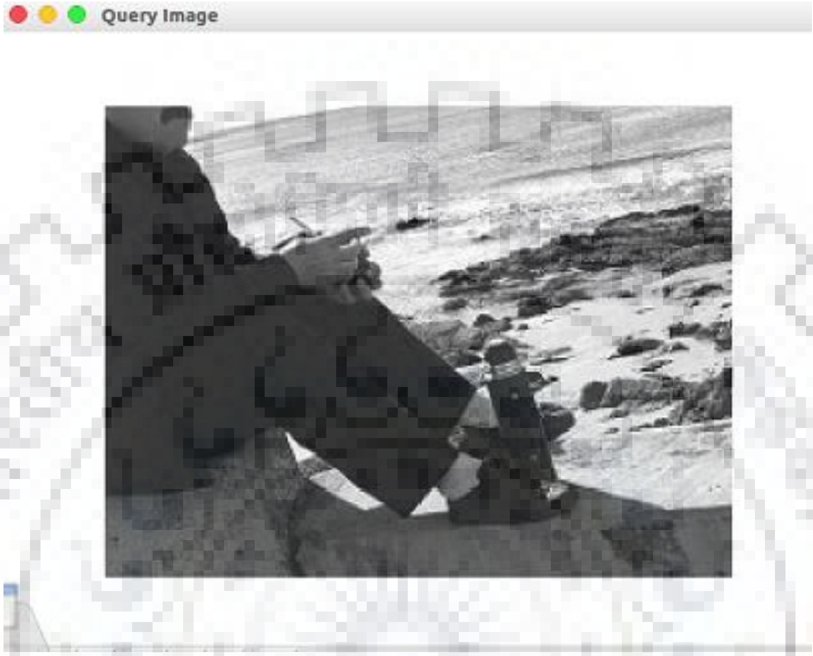
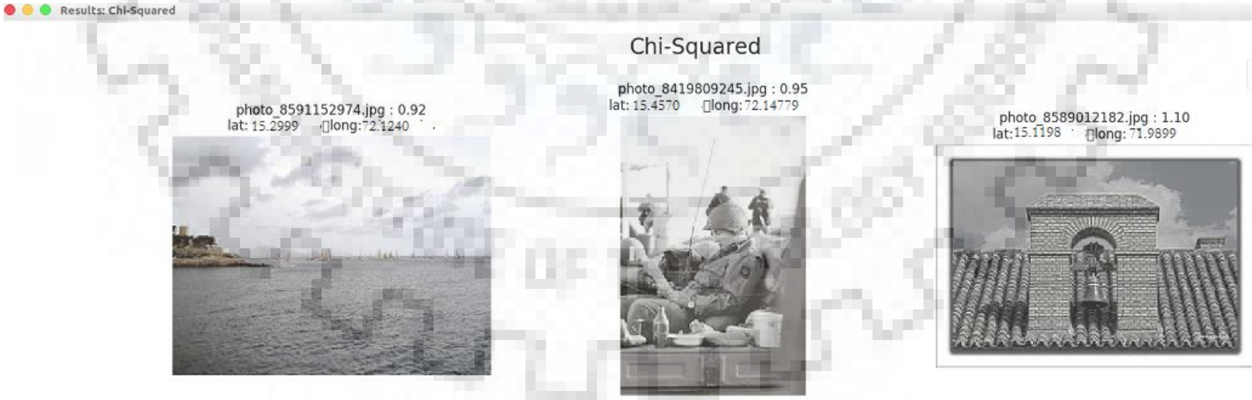


Figure 4.5Query Image

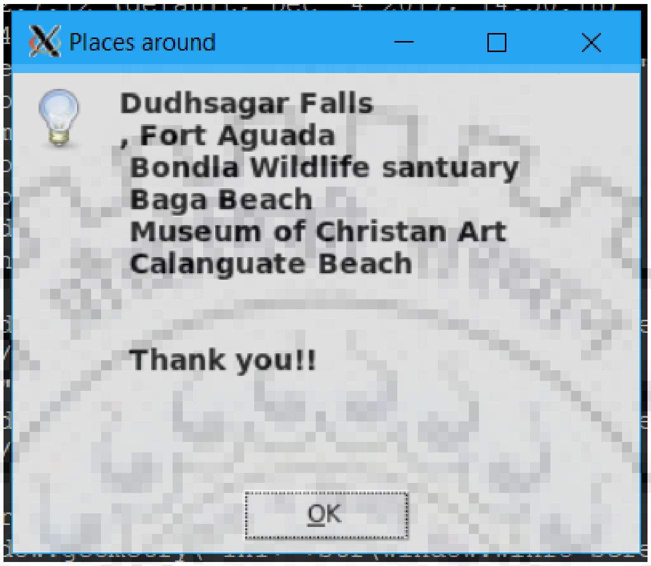
After executing the program the results for the given picture as shown in the figure 4.6.



```
Chi-Squared

photo_8591152974.jpg : 0.92
lat: 15.2999    long:72.1240

photo_8419809245.jpg : 0.95
lat: 15.4570    long:72.14779

photo_8589012182.jpg : 1.10
lat:15.1198    long: 71.9899
```

Figure 4.6 Result Images

These results shows the address for the query images as it can be clearly seen two of them are showing beaches. All of 3 images are from and near Goa beaches Results only shown from India as we given our location to the recommendation system. Apart from these recommendation using Api for information about the results have also provide additional information. Figure 4.7 shows the results on searching the GPS coordinates to find out about place.



Dudhsagar Falls
, Fort Aguada
Bondla Wildlife santuary
Baga Beach
Museum of Christan Art
Calanguate Beach

Thank you!!

Figure 4.7 Suggestions for the recommended results

For the next results an odd image is taken by us. We choose image of bird in order to find what recommender system will recommend to user on this image. Because Bird doesn't associate with any place. It generally associated with the hobby of bird watching.

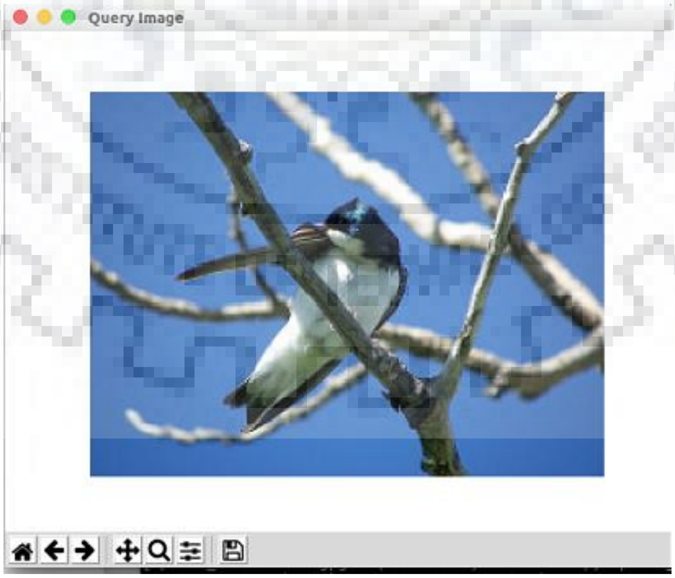


Figure 4.8 Query Image

The results associated with the above query image is shown in figure 4.9. This time we increased the no. of results to be shown because this was an odd test to check the recommender system capabilities.

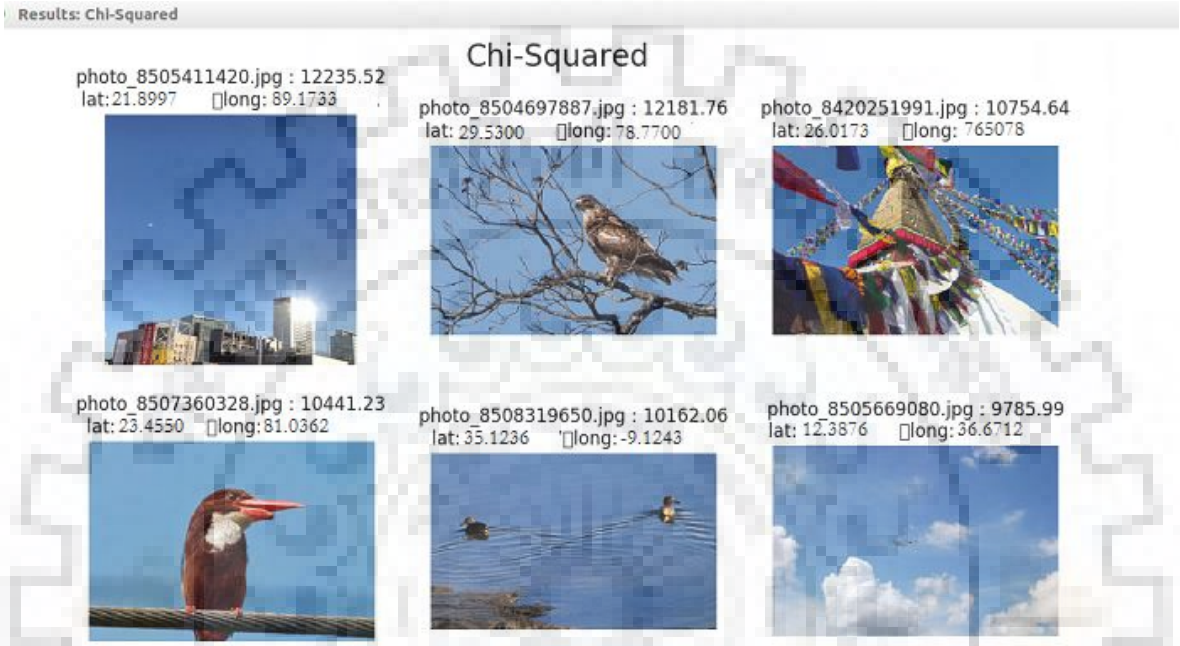


photo_8505411420.jpg : 12235.52
lat: 21.8997 □long: 89.1733

photo_8504697887.jpg : 12181.76
lat: 29.5300 □long: 78.7700

photo_8420251991.jpg : 10754.64
lat: 26.0173 □long: 765078

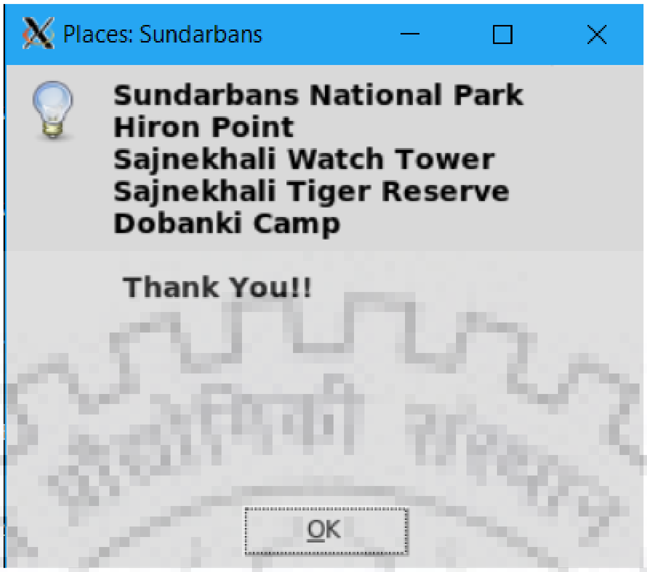
photo_8507360328.jpg : 10441.23
lat: 23.4550 □long: 81.0362

photo_8508319650.jpg : 10162.06
lat: 35.1236 '□long: -9.1243

photo_8505669080.jpg : 9785.99
lat: 12.3876 □long: 36.6712

Figure 4.9 Results images

This result shows that the recommender system was able to recognize the content of the image and tuned the results according to the query image. These places can be visited by the people to fulfill their hobby of bird watching. These photos are from various wild life centuries. Similarly with this result additional information about the place have been retrieved shown in fig 4.10



Sundarbans National Park
Hiron Point
Sajnekhali Watch Tower
Sajnekhali Tiger Reserve
Dobanki Camp

Thank You!!

Figure 4.10 Suggestions for the recommended result

4.3 Evaluation

To evaluate the performance of the recommender in comparison to the system proposed by Cao et al in [11] we are going to use Decision Support Accuracy Metrics [14]. This metrics view result as binary operation which tells whether the result is good or not.

We are using F measure to evaluate the performance of recommender on 5 most used queries in the survey.

$$F\ measure = \frac{2PR}{P + R}$$

Where P is Precision and R is Recall. No. of relevant item to the user out of the total recommendation is called Precision. Recall can be defined as the no. of relevant item out of the set of recommended item. Formula to calculate them is follows.

$$P = \frac{Correctly\ recommended\ items}{Total\ no.of\ recommendation}$$

$$R = \frac{Correctly\ recommended\ items}{Total\ no.of\ useful\ recommendation}$$

Using these metrics the comparison is performed between two recommender systems. Recommender system based on [11] is represented by R1 and the recommender system proposed by R2. The higher the value of the F measure the better the results for the

Queries	R1	R2
Beach	0.698	0.92
Mountains	0.59	0.721
Architecture	0.67	0.847
Island	0.47	0.59
Flower	0.63	0.91

Table 1 F-measures comparison

As we can see in the metrics table 1 results of R2 is better for every query. The reason for better F measure in comparison to R2 is better Recall. Our recommendation system improved the Recall using weather factor and distance calculation.

5. Conclusion and Future Works

In this dissertation a methodology to develop a dynamic recommender system is proposed. This system depends on the representative tags and images to recommend the places to visit. The system overview is showed in figure 1. User have the choice to put query either in form of tag or in image. This system looks for the similarity between user queries and representative images of the clusters. On the basis of that similarity we get potential results to show user. The reachability for the location is prioritized. Closer location to the user will be recommended. Before displaying results to user those results pass through the dynamic module i.e. weather check. After calculating the score of each image according to the current weather. These images are sorted according to the score and displayed to the user. After this information about the recommended place are searched for e.g. Places to visit in that area and finally all of this shown to the user.

Including weather factor and reachability in recommender system have certainly improved the suggestions but more sophisticated method are required to deal with other dynamic factors.

5.1 Future Works

As we've seen that this approach had improved the suggestion but still it is not very efficient. Partial reason for that is non uniformed distribution of the usage of Flickr. In America and European region Flickr is vastly used. So we have almost every tourist destination covered in those regions. And in places like middle-east and Africa some of the tourist destination is yet to be covered. This will be covered over time.

There are various areas where efforts are still need to be put on. This recommender system is a generalized system. This could be made personalized recommender system using the attributes of the end user. Finding user interest would be helpful for the system to provide better suggestion to the user. For example location can be recommended considering the gender and age of the end user. Another area to work on is to provide suggestion after learning user's activity which will make it even more personalized to the user. Increasing the dataset will also improve the results.

6. References

[1] Smith, Craig. “Flickr Stats.” *DMR*, 6 May 2018, expandedramblings.com/index.php/flickr-stats

[2] Yahoo! TRAVEL, http://www.travel.yahoo.com/

[3] Google Maps, http://maps.google.com/

[4] Flickr, http://www.Flickr.com

[5] Takeuchi, Y., & Sugimoto, M. (2006, September). CityVoyager: an outdoor recommendation system based on user location history. In *International Conference on Ubiquitous Intelligence and Computing* (pp. 625-636). Springer, Berlin, Heidelberg.

[6] Cheng, A. J., Chen, Y. Y., Huang, Y. T., Hsu, W. H., & Liao, H. Y. M. (2011, November). Personalized travel recommendation by mining people attributes from community-contributed photos. In *Proceedings of the 19th ACM international conference on Multimedia* (pp. 83-92). ACM.

[7] Zheng, V. W., Zheng, Y., Xie, X., & Yang, Q. (2010, April). Collaborative location and activity recommendations with GPS history data. In *Proceedings of the 19th international conference on World wide web* (pp. 1029-1038). ACM.

[8] Zheng, Y., & Xie, X. (2011). Learning travel recommendations from user-generated GPS traces. *ACM Transactions on Intelligent Systems and Technology (TIST)*, *2*(1), 2.

[9] Mazimpaka, J. D., & Timpf, S. (2016). Trajectory data mining: A review of methods and applications. *Journal of Spatial Information Science, 2016*(13), 61-99.

[10] Mathew, W., Raposo, R., & Martins, B. (2012, September). Predicting future locations with hidden Markov models. In *Proceedings of the 2012 ACM conference on ubiquitous computing* (pp. 911-918). ACM.

[11] Cao, L., Luo, J., Gallagher, A., Jin, X., Han, J., & Huang, T. S. (2010, March). Aworldwide tourism recommendation system based on geotaggedweb photos. In *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on* (pp. 2274-2277). IEEE.

[12] Accuweather, www.accuweather.com

[13] OpenWeatherMap API www.openweathermap.org

[14] Isinkaye, F. O., Folajimi, Y. O., & Ojokoh, B. A. (2015). Recommendation systems: Principles, methods and evaluation. Egyptian Informatics Journal, 16(3), 261-273.

[15] Borràs, J., Moreno, A., & Valls, A. (2014). Intelligent tourism recommender systems: A survey. *Expert Systems with Applications*, *41*(16), 7370-7389.

[16] Fukunaga, K., & Hostetler, L. (1975). The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Transactions on information theory*, *21*(1), 32-40.

[17] Yang, J., Yu, K., Gong, Y., & Huang, T. (2009, June). Linear spatial pyramid matching using sparse coding for image classification. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on* (pp. 1794-1801). IEEE.

[18] Pliakos, K., & Kotropoulos, C. (2014, May). Simultaneous image tagging and geo-location prediction within hypergraph ranking framework. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*(pp. 6894-6898). IEEE.

Thesis

ORIGINALITY REPORT

6%	3%	4%	0%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Liangliang Cao, Jiebo Luo, Andrew Gallagher, Xin Jin, Jiawei Han, Thomas S. Huang. "Aworldwide tourism recommendation system based on geotaggedweb photos", 2010 IEEE International Conference on Acoustics, Speech and Signal Processing, 2010 Publication	2%
2	www.microsoft.com Internet Source	1%
3	link.springer.com Internet Source	1%
4	Kurashima, Takeshi, Tomoharu Iwata, Go Irie, and Ko Fujimura. "Travel route recommendation using geotagged photos", Knowledge and Information Systems, 2013. Publication	1%
5	summit.sfu.ca Internet Source	<1%
6	Borràs, Joan, Antonio Moreno, and Aida Valls. "Intelligent tourism recommender systems: A	<1%