



# Travel Route Recommendation System using User Keyword Search

Ashishika Singh, S. Babu

**Abstract --** Travel and tourism is a field, which have been growing substantially over the past few decades. The competitiveness in marketing and need of fulfilling customer experience in travel have given many opportunities for today's technological advancements to play a crucial role in it. Those technology aspects are Big Data and Data Mining. Data Mining uses technologies of statistics, mathematics, machine learning and artificial intelligence. It aims to classify original, valid, useful, potentially and understand correlations and patterns. Data mining with the help of Big Data - Hadoop can help analyze and derive information, which can increase the growth of industry and give accurate suggestion to customer. The reason of combining capabilities of Hadoop is it can handle all sorts of data such as Structured or Unstructured. The main objective of this project also revolves around the same principle giving the best Customer Experience. By combining the power of Data Analytics of data mining, Big Data and programming capabilities of Java, this project focuses on building a customer centric Keyword Aware Travel Route Framework.

**Keywords:** Travel and tourism, Big Data, Hadoop Data Mining, Structured Data, Unstructured Data LBSN, Keyword aware travel route.

## I. INTRODUCTION

With the widespread of social media platforms such as Facebook, Flickr, and Twitter etc. people are online on a very large scale these days. It is now possible for the general masses to provide their check in records, photographs and their whole experience while travelling very easily. The more and more people share about their travel experiences on social networks, the larger the data pool of travel related information grows. One can easily look into one another's travel experiences and gain valuable insights for their own travel plans. Also, as this huge dataset is readily available to anyone at just a click away, people tend to get confused on their personal preferences with so many options to choose from and make this a very tedious work for themselves.

By leveraging all of this diverse user travel data we can incorporate more number defining characteristics for the POIs (Point of Interests) in the datasets to automate the complete process of trip planning with more personalized and user centric trip offerings.

Hence, here, we propose a proficient Travel Recommendation system based on Keyword search that utilizes information extraction from people's historical records and social life.

## II. EXISTING SYSTEM

Currently there are many systems available for travel and tourism, which assist us just in building our itinerary. Our existing system is utilizing collaborative filtering technique to give recommendation based on user's profile.

There are two main components of this existing system:

- Point-of-Interest Discovering and Ranking
- Travel Route Planning

### A. Point-of-Interest Discovering and Ranking<sup>1</sup>

The first step in generating the travel package is to obtain a set of POIs that the users might be interested.

1. Point-of-Interest Discovery: The POIs that the users might visit are time-dependent. To discover candidate POIs, we divide the travel duration into several times. Then, we can find the preferred type of locations for each period, according to the user's profile. Based on the check-in records in a certain period, people tend to visit the same.

2. Point-of-Interest Ranking: For each time period, we select the appropriate POIs from the discovered POI set as the candidate locations for route planning. We rank these POIs according to user preference. The challenge is that there is no check-in record for most of the POIs for a particular user; hence, the preference of locations cannot be gained in these records directly. Users with the same tastes in traveling may have similar travel records. Thus, we can leverage the collective intelligence of crowdsourced data to reveal the preferred locations, which are totally new for users.

### B. Travel Route Planning

A set of POIs in a travel package without order is not complete as traveling is spatial-time-dependent. Thus, we need to determine the visiting routes with the recommended POIs by taking into account the user's spatio-temporal constraints.

Manuscript received on February 10, 2020.

Revised Manuscript received on February 20, 2020.

Manuscript published on March 30, 2020.

#### \* Correspondence Author

Ashishika Singh, Department of Computer Science and Engineering, SRM Institute of Science and Technology, Chennai, India. Email: ashishikasingh2014@gmail.com

S. Babu, Department of Computer Science and Engineering, SRM Institute of Science and Technology, Chennai, India. Email: babus@srmist.edu.in

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>)

**Limitations of this Existing System**

- Point-of-Interest Discovery is totally based on existing profile of the user and what places he have already visited, for new user this would be a complex task to determine POIs for him as his profile is new and he might not have any preference present in his profile.
- This system is solely based on POIs as per different users visiting time, whereas the proposed system will take account of all user preferences new or old through Keyword matching.

**III. PROPOSED SYSTEM**

Creating a system where in we already gather historical data of user travels via different modes such as user reviews, user tags, check in information and social media. After which when a new user enters keywords to search, which can be directly a name of a place or description of a place such as white sand beaches etc. The system will effectively match those keywords to the entire user reviews present in the system, match the keywords to designated POIs and then generate the most suitable and best travel route for complete user satisfaction.

These travel routes recommendations will be based on the following set of parameters:

- Level of Attractiveness of the main attractions that we pass through.
- Visit these attractions at their said arrival time
- Also based on various social media influencer movement such as travel blogger etc.

Modules for Implementation:

1. User Registration.
2. Geo specific Keywords.
3. Temporal Keywords.
4. Candidate Route Generation.
5. Travel Route Exploration
6. Product recommendation

- **User Registration:** User will register their details like name, user id, password, email-id, contact, address and other details. Those details will be registered through Graphical User Interface (GUI). That information will be stored on database like MySQL.

- **Geo-Specific Keywords:** Some tags are specific to a location, which represents its spatial nature. To identify such Keywords for ranking the POIs we utilize the MapReduce Algorithm of Bigdata – Hadoop. The MapReduce algorithm comprises of two important functions, namely Map and Reduce. The Map function takes a set of data and converts it into another set of data, where each element is further divided into tuples (key/value pairs). Secondly, the reduce function, which takes as an input the output from a map and combines those data tuples into a smaller set of tuples. Here we will try to identify if the keyword falls under Geo – specific keyword or not. For that, geo terms or geographical characteristics are to be identified. We make,

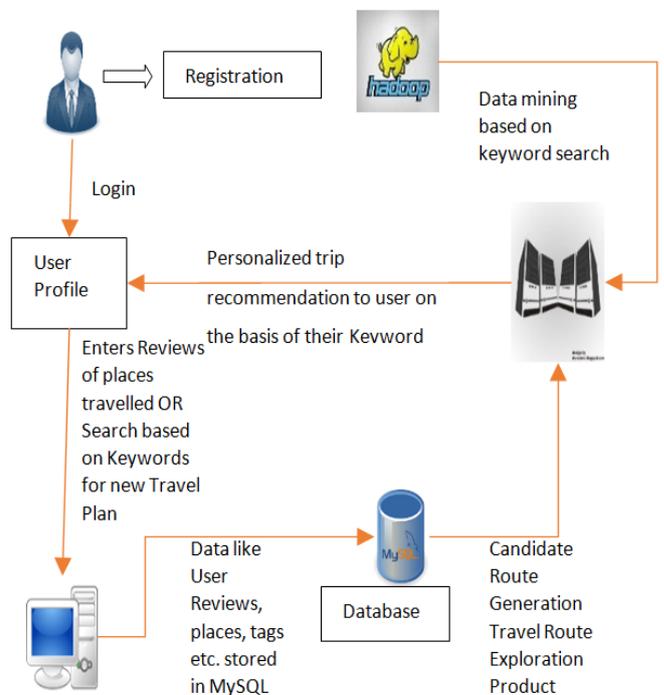
Key: Value pairs, such as  
Geo specific: Mumbai

- **Temporal Keywords:** Some tags are defined by its time interval, which represents its temporal specific nature. As discussed, we use MapReduce of Hadoop here too, to find those temporal keywords.

- **Candidate Route Generation:** In this section, we talk about generation or finding of candidate routes to diversify and increase our route dataset. In this algorithm, the goal is to match characteristics of POIs with the entered unchanged texts and set our mining filters in existing routes accordingly. However, as observed it is not completely unnatural that sometimes our dataset for route might not fulfill all the query criteria or might not be strongly related to the query keywords. Hence this candidate route generation algorithm helps in merging various other routes and diversify our dataset as well as increase the volume of it. The sub – series of courses form the newly constructed candidate routes.

- **Travel Route Exploration:** As we discussed in previous section about increasing and diversifying our route dataset, in this section using that dataset we set out to give one final, most optimal route that connects some or all key words which are given by user. First, we understand and apply the function of matching to user query to process it correctly. Then for the next step we apply the knowledge or logic behind skyline query in order to justify which route is most suitable for the recommendation of travel routes.

- **Product recommendation:** People post their review related on products that is famous on the particular location. With their travel experience they will post information related to the product. Our system will suggest the famous product for the upcoming people. So in future people who are all like to go to that they can easily get their need. These products may include local cuisine, handlooms, Textiles/ Clothing, Jewelry or even famous food joints not just dish, etc. These will also be quantified based on the user preference or interests and then rated.



**Fig. 1. System Architecture**

#### IV. RELATED WORK

##### A. Keyword Analysis

The below mentioned measures should be taken so as to lead an Effective Keyword Analysis

Databases to be searched for the following keywords: tourist, big data, Hadoop, tourism, destination, travel, hotel and data mining.

As numerous subsidiaries of words like tourist, travel and vacation exists, lexicon for comparative words ought to likewise be incorporated into the examination of keywords.

##### B. Effective Keyword Analysis

Some of the regions where we can find the use of data mining actively are like predictive analysis or forecasting, leisure industry or vacation management industry, recommendation applications for tourism.

Keyword Analysis helps in creating explicit Keyword oriented framework, which spotlights on keywords of most significance in that field. Similarly, [1], [5] helps in developing personalized travel routes based on user queries. For example, [1] constructed a system for time sensitive routes based on location popularity, proper visiting time and visiting order. Henceforth it can help target the travel industry segment for customer driven suggestions.

The various sectors of tourism where this can be applicable are:

- Most famous attractions
- Local cuisines
- Local products
- Hotels, Activities
- Weather and Climate suitability.

In [5], routes were ranked based on the attractiveness of the location, proper visiting time and distance to query location. Unequivocally, a module for extraction of keywords developed by us to gather and organize all the labels related to POI in order to match it with input keywords with a good success rate. Additionally, planned a candidate route creation algorithm to give us candidate routes as output, which may satisfy all the parameters.

- **Geo-Specific Keywords:** Here we will try to identify if the keyword falls under Geo – specific keyword or not. For that as mentioned in earlier geo terms or geographical characteristics are to be identified and quantified i.e. if the rating of that keyword or Geo Specificity crosses an established threshold it will be considered as Geo Specific Keyword.

- **Temporal Keywords:** The quantification of temporal keywords is similar to Geo specific keyword where the keywords are characterized according to time and rated. After that if these keywords cross the initial threshold, then only they are classified as Temporal keywords.

The task of location recommendation [2], [3], [7] is to recommend new locations that the user has never visited before. For the location recommendation part, [2] directed the people who tend to visit nearby locations but maybe interested in more distant locations. [3] recommended a list of POIs for the user to visit at a given time by exploiting

both geographical and temporal influences. [7] focused on the relationships between individuals and recommended the locations that influential users have been to. [8] defined a similarity function for measuring how well a trajectory connects to query locations.

##### C. Candidate Route Generation

**Pre-processing:** After the Query Keyword Matching check, all the input routes must have at least one user-specific keyword. Let  $T_i$  be the trajectory, we now know that it contains series of POIs,  $(P_1$  to  $P_n)$ . We finalize a data structure in form of (head, tail) in order to trace back our complete route in single – single step switch from one POI to another, i.e.,  $\{P_1$  to  $P_2, P_2$  to  $P_3, P_{n-1}$  to  $P_n\}$ . We created dictionary array for both headset and tailset in order to store information of head as well as tail in their respective dictionaries.

All the points which are common and are combines needs to be in order with time: This means that it is more informative to merge  $(P_i, P_j)$  &  $(P_k, P_l)$  if both  $P_j$  and  $P_k$  turns to be identical places. Apart from the apparent distance between the two points, there is one other major factor that needs to be considered is the time of visit order among the common points. Time of visit for  $P_k$  should be more than the time of visit for  $P_i$  as time of tail should always be greater than time of head, for us to substitute  $P_j$  with  $P_k$ .

To get all the possible travel routes we proposed and designed a methodology, which utilizes Depth-First Search approach. **Data Pre-Processing:** After Query Keyword Matching check, all the input routes must have at least one user-specific keyword.

##### D. DFS-based route enumeration

Since it is required to gather all of the various possible tracks from the original tracks, we reconsider the original track and remake the new tracks by merging the (head, tail) dictionaries using common and combined points. The approach here is based on Depth-First search methodology. POIs present in headset are to be treated as source, and before backtracking we try to traverse all the way as much possible alongside each of the present links. Furthermore, it is relatively straightforward to consider the time-period query in the process of DFS. In Algorithm 2, we only need to add a stricter time-query limitation to line 7, e.g., the size of  $S$  should not exceed the number of POIs in a route threshold.

**Algorithm:** Candidate Route Generation

Input: Raw trajectory set  $T$  ;

Output: New candidate trajectory set  $T_c$

- 1: Initialize a stack  $S$ ;
- 2: Split each route  $r \in T$  into (head,tail) subsequences;
- 3: Reconstruct(headSet).
- 4: Procedure Reconstruct(Set):
- 5: foreach (head,tail)  $\in$  Set do
- 6: endFlag = False;
- 7: if  $S$  is empty or tail.time >  $S$ .pop().time then
- 8: Push head in  $S$ ;
- 9: Push tail in  $S$ ;
- 10: else
- 11: Push head in  $S$ ;
- 12: endFlag = True;

- 13: if endFlag is False then
- 14: Reconstruct(tailSet)
- 15: Insert S in Tc;
- 16: Procedure End Procedure End

**D. Travel Route Exploration**

In this segment, we present the algorithm of trajectory reconstruction and how to apply Skyline query for our online suggestion framework.

Besides, we propose an inexact algorithm to accelerate the continuous Skyline question. In the below mentioned algorithm, we try to reflect the underlying approach for travel route exploration module.

**Algorithm:** Travel Route Exploration

Input: Below are the mentioned inputs:

Keyword set K, range of query Q, a user U;

1: Initialize priority queue CR, KRT;

2: Scan the database once to find all candidate routes covered by region Q;

/\* Fetch POI scores and check keyword matching

3: foreach route r found do

4: r.kmatch 0;

5: foreach POI p 2 r do

6: r.kmatch r.kmatch + KM(p,k);

7: if r.kmatch then

8: Push r into CR;

/\* Initialize an arbitrary skyline route,

9: CR.r0 route r with the largest value of an arbitrary dimension;

/\* Greedy algorithm for representative skyline\*/

10: KRT I-greedy(CR);

11: return KRT.

**V. RESULTS**

As mentioned earlier, we classified the keywords into geo specific, Temporal and Product Attribute Keywords. Further after classification we ranked those keywords, below table shows the precision factor of deciding the keywords in those categories. In Geo-specific module we ranked the tags, same for temporal keywords etc., we then use those ranks to calculate precision@K.

**Table- I: Precision Values for Keyword Extraction**

	P@10	P@20	P@30
Location based Tags	1.000	1.000	0.965
Time based Tags	0.912	0.698	0.719
Product Credit	1.000	0.880	0.699

Just to give high-level knowledge about the social network dataset used for the project below mentioned are some of the specifics of the dataset

**Table- II: Specifics of Location Based Social Networks**

Characteristics		Social Platform	
		Facebook	Flickr
# Records	Check - in	9,10,024	5,22,104
# Nodes	User- Profiles	32,648	6,019

	Point of Interests	1,64,920	1,01,950
# Edges	Friend	41,853	34,714

In the dataset some keywords were highly specific to some places where as some general. For Example, the Geo specific keyword “Leh-Ladakh” is place in Jammu and Kashmir but is also highly associated with the Temporal Keyword “winters” as it has very cold weather. Similarly, Sunset is obviously a Temporal Keyword but can be associated with many different places especially hilly areas such as The Ridge in Shimla or Darjeeling etc. Below mentioned are some of the top results of Keyword Extraction.

**Table- III: Top Results of Keyword Extraction**

S.NO	Keyword Types		
	Location Specific	Time Specific	Product Credit
1.	Goa	Sunset	Water Sports
2.	Leh Ladakh	Winters	Wool cloth
3.	Taj Mahal	Full Moon Night	Sweets
4.	The Ridge	Sunset	Tibetan Cloths
5.	Darjeeling	Sunrise	Tea
6.	Red Fort	Fireworks	Clothing
7.	Rann of Kutch	Springs	Rannotsav
8.	Kerala House Boat	Summer	Banana Chips
9.	Victoria Memorial	Historic	Bengal Sweets
10.	Mewar Fort	Summers	Ghewar

In this part, using the Photographic Tags and POI datasets, we will highlight the evaluation criteria, and therefore the baselines for evaluation. We utilized the Facebook and other LBSNs dataset equaling 1,10,926 photos. We manually matched the photo data with 403 attractions.

For estimating the execution of the extraction check in data, we consider a labeled photo as a Real Time check-in photo {who: user ID; where: labeled POI; when: photographed time}. On the basis of Real Time check-in photo, we calculated evaluation measures, precision, recall, and F1 score by considering the factors such as check in method, manually labeled photos on specific POIs.

With calculate these evaluation measures for two baseline extraction methodologies and three proposed extraction methodologies. Below are the mentioned evaluation measures.

$$\begin{aligned}
 \text{precision} &= \frac{\sum_{p \in L} |I_p^{GT} \cap I_p^m|}{\sum_{p \in L} |I_p^m|} \\
 \text{recall} &= \frac{\sum_{p \in L} |I_p^{GT} \cap I_p^m|}{\sum_{p \in L} |I_p^{GT}|} \\
 F1 &= \frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}
 \end{aligned}$$

where IGTP is a set of physically marked photographs on POI p and Im p is a set of photographs named with p by a check in extraction technique m. We perform a 2-fold approval: Each 50% of the ground truth is utilized as preparing information and test information, individually. As candidates for the check in extraction technique m, we present the accompanying two baseline extraction strategies, and our three proposed extraction methods.

- **Base:** A standard strategy that only takes duplicate/near duplicate photograph groups with an official POI name.
- **Base+:** A standard strategy that takes about duplicate/near duplicate photograph groups with numerous POI names removed by a best in class name development technique.
- **SCE:** A basic, Synonym-based Check-in Extraction. That is one POI can have many synonyms and we identify them.
- **CCE:** A component, Collective Check-in Extraction. This done once we get the synonym-based check in and then try to find clusters of check in which have the same type.
- **SCE + CCE:** Our proposed method combining the two components.

**Table- IV: Performance stats for Extraction of Check in Data**

	Precision	Recall	F1 Score
<b>Base</b>	0.949	0.429	0.579
<b>Base+</b>	0.937	0.512	0.681
<b>SCE</b>	0.931	0.615	0.726
<b>CCE</b>	0.942	0.457	0.634
<b>SCE + CCE</b>	0.917	0.696	0.791

## VI. CONCLUSION

We propose a system, an architecture in which people can give a lot of keyword input and an inquiry region, and for which query results contain different travel routes. We propose a novel keyword extraction module to separate three kinds of keywords: Geo-specific keywords, Temporal keywords and Attribute keywords. We separate three travel elements of POIs from LBSNs: appeal of POIs, the visiting time of POIs and the local social impact. We propose a trajectory reconstruction strategy to segment directions into fragments by thinking about space and time related features. Skyline query for trip course search is received to join the multidimensional estimations (POI engaging quality, appropriate visiting time and topographical social impact) of courses, which increment the assorted variety of the prescribed outcomes. An approximate algorithm is embraced to determine effective outcomes for the online intuitive framework.

## REFERENCES

1. H.-P. Hsieh and C.-T. Li, "Mining and planning time-aware routes from check-in data," in Proc. 23rd ACM Int. Conf. Conf. Inf. Knowl. Manage., 2014, pp. 481–490.
2. M. Ye, P. Yin, W.-C. Lee, and D.-L. Lee, "Exploiting geographical influence for collaborative point-of-interest recommendation," in Proc. 34th Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval, 2011, pp. 325–334.

3. Q. Yuan, G. Cong, and A. Sun, "Graph-based point-of-interest recommendation with geographical and temporal influences," in Proc. 23rd ACM Int. Conf. Conf. Inf. Knowl. manage., 2014, pp. 659–668.
4. V. S. Tseng, E. H.-C. Lu, and C.-H. Huang, "Mining temporal mobile sequential patterns in location-based service environments," in Proc. Int. Conf. Parallel Distribute. Syst., 2007, pp. 1–8.
5. W. T. Hsu, Y. T. Wen, L. Y. Wei, and W. C. Peng, "Skyline travel routes: Exploring skyline for trip planning," in Proc. IEEE 15th Int.Conf. Mobile Data Manage., 2014, pp. 31–36.
6. Y. Zheng, L. Zhang, X. Xie, and W.-Y. Ma, "Mining interesting locations and travel sequences from GPS trajectories," in Proc.18th Int. Conf. World Wide Web, 2009, pp. 791–800.
7. Y.-T. Wen, P.-R. Lei, W.-C. Peng, and X.-F. Zhou, "Exploring social influence on location-based social networks," in Proc. IEEE Int. Conf. Data Mining, 2014, pp. 1043–1048.
8. Z. Chen, H. T. Shen, X. Zhou, Y. Zheng, and X. Xie, "Searching trajectories by locations: An efficiency study," in Proc. ACM SIGMOD Int. Conf. Manage.



## AUTHORS PROFILE

**Ashishika Singh**, is currently pursuing Master's Degree in Computer Science and Engineering at SRM Institute of Science and Technology, Kattankulathur, Chennai, India. She pursued her Bachelor's from Amity University, MP. Her areas of interests are Data Analytics, Data mining and DBMS.



Metrics.

**Dr. S. Babu**, is an associate professor of Computer Science and Engineering at SRM Institute of Science and Technology, Kattankulathur, Chennai, India. He earned his PhD in CSE from Anna University, Chennai, India. He has over 13 years of experience in Teaching and Research. His areas of interests are Internet of Things and Software