Trip Planning Using Hybrid Recommendation Systems

[1]S. Carvalho, [2]A. Correia, [3]D. Kammu
[1,2,3] Department of Information Technology, Fr. Conceicao Rodrigues College of Engineering, Mumbai, India

Abstract : Planning a travel itinerary to an unfamiliar destination requires a lot of work and effort to design a desirable travel plan. Especially nowadays, with the amount of tourism information available on the internet, it is really difficult to curate a trip plan from the plethora of choices given to you online. Unlike most travel recommendations systems which recommend itineraries based on user ratings, in this paper we seek to solve the problem using a hybrid recommendation algorithm that is based upon collaborative and content-based filtering algorithms. The hybrid algorithm combines recommendations of both the filtering algorithms to get the best results and in doing it so also overcomes many of the disadvantages possessed by both the algorithms, as the disadvantage of one is overcome by an advantage of the other. The goal of the research presented in this paper is to help tourist overcome the complex task of Identifying, Organizing and planning a trip that comprises popular and interesting points of interest. The system offers personalized recommendations to the users based on their interests and travel history, and other metadata of a given location.

***Keywords*— Recommendation Systems, Collaborative Filtering, Content Based-Filtering, Travel, Point of Interests.**

I. INTRODUCTION

These days tremendous amounts of tourism information is available over the web. This information can prove to be very useful to a user who wishes to plan a trip to a location of which he has no prior knowledge. However, it is often time consuming and overwhelming to look through such volumes of data and find relevant points of interest and curate a well-defined tour plan. Therefore, in this paper we seek to filter out unnecessary information and extract out personalized recommendations based on the user's preferences using a hybrid recommendation algorithm.

Given the complexity of this theme, tourism is a privileged area for the application of artificial intelligence, and, in particular, Recommendation Systems (RS). RSs devise the use of computational means to calculate a great number of decision components that the human brain can't integrally assimilate, giving users the result that they expect. Recommendation systems are the heart of almost every internet business today, from Amazon to YouTube to Facebook. Providing good recommendations whether its friends, movies or travel plans, goes a long way in defining the user experience and enticing your customers to use your platform. Recommender Systems can be roughly defined as information filtering and decision support tools that provide products and services that match user preferences [1-2]. Most recommendations systems use the collaborating filtering algorithm to filter the results and prevent information overload [3]. This approach poses the cold start problem in which giving recommendations to novel users who have no preference on any items, or recommending items that no user of the community has seen yet becomes a very difficult task. In the case of recommendation systems in the tourist domain, generally the users seeking recommendations are new users that don't have any previous visits to a similar location.

This problem leads us to propose a hybrid recommendation method that combines collaborative and content-based filtering algorithms. Our research develops a cold-start aware system that takes into consideration user preferences such as number of days of visit, pace of visit, and similar locations the user has visited to give recommendations to the user to plan his trip.

II. RELATED WORK

Zhiwen et al. implements a method of obtaining the digital footprints of users using location aware information to infer user preferences and recommend multiple point of interest (POI) [4]. Z. Bahramian et al introduces a hybrid system that combines an artificial neural network with case-based reasoning to offer personalized tours to the users based on their preferences and some contextual information [1]. Wen et al. proposed a keyword-aware travel recommendation framework using knowledge extraction from users' historical social interactions and mobility records combined with a keyword extraction module to classify POI-related tags that match with query keywords [5].

III. METHODOLOGY

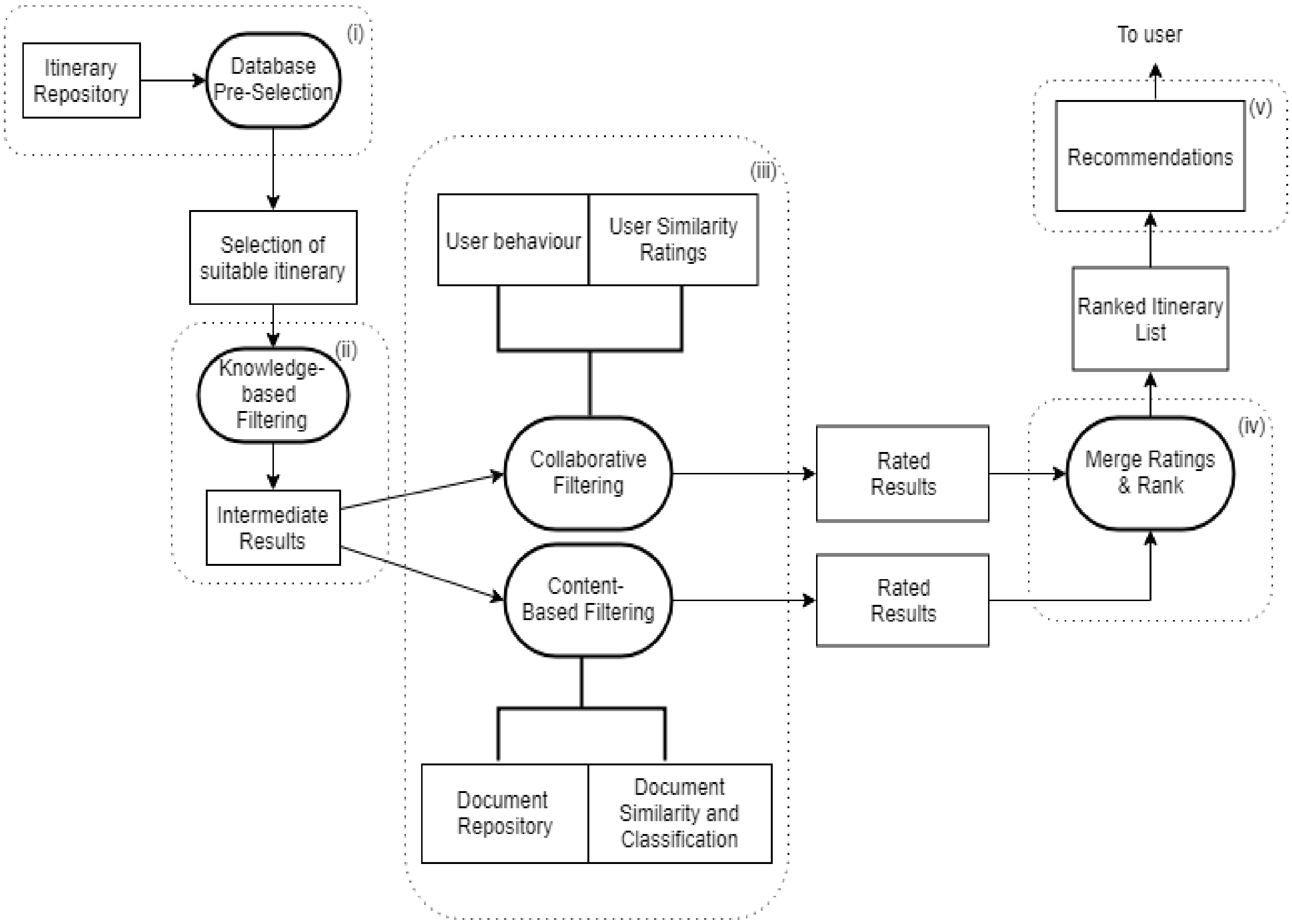

Figure 1. Hybrid Recommendation System Architecture

Modules of the system as per Figure 1:

(i) Database Pre-Selection

In the Database (DB) pre-selection phase we retrieve an initial selection of items using simple DB interactions. We retrieve only those fields relevant to the recommendation from the data-set.

(ii) Knowledge-based Filtering

Using knowledge based filtering based on knowledge of the location, number of days, and pace of travel we further reduce the size of the item set. We obtain a very efficient reduction of the dataset at a very early stage of the workflow.

(iii) The Recommendation Engine

We utilize recommendation algorithms such as collaborative filtering and content based filtering to extract item ratings for every item that is to be recommended.

Content-Based Filtering

Content-based filtering systems do not require data of past user activities like collaborative filtering. They provide recommendations based on user profiles and metadata it has on particular items [6].

The models we are building are to compute pairwise similarity between metadata of POIs. To numerically quantify the similarity between the metadata as mathematical quantities, we represent the POI metadata as vectors, i.e. every POI is depicted as a series of numbers, where each number represents a dimension and n is the size of the vocabulary of all the POIs put together. Here, we have used two popular vectorizers those are CountVectorizer and TF-IDFVectorizer.

CountVectorizer computes the size of the vocabulary. The vocabulary is the number of unique words present across all POIs. It is a common practice to not include extremely common words such as a, the, is, had, my, and so on (also known as stop words) in the vocabulary [7].

TF-IDFVectorizer (Term Frequency-Inverse Document Frequency) takes the aforementioned point into consideration and assigns weights to each word according to the following formula. For every word i in POI j, the following applies:

$$w_{i,j} = tf_{i,j} \times log\,(\frac{N}{df_i})$$

In this formula, the following is true:

1. $w_{i,j}$ is the weight of word i in document j
2. df_i is the number of POIs that contain the term i
3. N is the total number of POIs

The weight of a word in a POI is greater if it occurs more frequently in that POI and is present in fewer documents. Another advantage of TF-IDF is that it speeds up the calculation of the cosine similarity score between a pair of documents. The weight $w_{i,j}$ takes values between 0 and 1 [7].

Cosine Similarity

The cosine similarity score computes the cosine of the angle between two vectors in an n-dimensional space. When the cosine score is 1 (or angle is 0), the vectors are exactly similar. On the other hand, a cosine score of -1 (or angle 180 degrees) denotes that the two vectors are exactly dissimilar to each other.

The cosine similarity score between two POIs, x and y, is as follow:

$$cosine(x, y) = \frac{x . y^T}{||x||.||y||}$$

Once we get a list of cosine similarity scores for that particular location with all other locations. We convert this into a list of tuples and sort this tuples on the basis of the cosine similarity scores. From this list we take the top 25 elements ignoring the first element as it refers to the similarity score with itself. These top 25 elements are the required result for the recommendation system.

Collaborative Filtering

Collaborative filtering is a system of predicting the future preference of a set of items for the user and recommend the items that will best match the user's preferences. This system leverages the power of community reviews and ratings to provide recommendations [8].

We will build our collaborative filtering model in context of Point-of-Interests. We use user-based collaborative filtering in which we find users similar to a particular user and then recommend POIs that those users have liked to the first user [8].

First we build a ratings matrix where each row represents a user and each column represents a POI. Then we compute the similarity between users from the ratings matrix using Cosine Similarity to construct a cosine similarity matrix. With the user cosine similarity matrix in hand, we now can efficiently calculate the weighted mean scores for this model. We use weighted mean as it makes more sense to give more preference to those users whose ratings are similar to the user in question than the users whose ratings are not. Using weighted mean enables us to get more relevant recommendations [2].

(iv) Hybrid Recommendation

For dealing with a hybrid recommendation system that uses more than one rating for every item, we now merge these multidimensional ratings into a single value using the mean value.

Hybrid Recommenders

Hybrid recommenders are powerful systems that combine the results of different types of recommendation algorithms. Every algorithm comes with its inherent set of advantages and disadvantages, and hybrid systems combat the disadvantages of one algorithm with the advantage of another [9].

We have built our recommender based on Content based filtering and Collaborative Filtering, since both of the recommenders suffer from one shortcoming or the other. The major shortcomings being the cold start problem of collaborative filters and the novelty problem associated with content-based filters. Thus to overcome these shortcomings, we build a robust system that combines both the models.

The workflow of our hybrid model will be as follows:

1. Take in a POI name and user as input
2. Use a content-based model to compute the 25 most similar POI's
3. Compute the predicted ratings that the user might give this 25 POI's using a collaborative filter.
4. Return the top 10 POI's with the highest predicted ratg.

(v) Recommendation Results

Finally we select 10-items with the highest score, based on which we return the final set of recommendations.

IV. RESULTS AND DISCUSSION

The algorithm is run on a dataset obtained from Kaggle titled "Point of Interest POI Database" created by Evan Hallmark. The dataset contains roughly 400,000 unique points of interest, including latitude and longitude, as well as other metadata required for providing recommendations [10].

E.g. If the user visited a POI called "Bire Wala Jattan" the recommendation engine would recommend POIs to him based on his visit to that location. In the scenario our recommendation system provides him with the recommendation given in Table 1 with POIs similar to his previous visit.

Table 1. Recommendation Results

16841	TANDIAN
18357	CHHAPIAN WALI
3353	RAUNI, INDIA
17437	DHAK PANDORI
3573	FATEHGARH GUJJRAN (LUDHIANA EAST)
347	CHAK KALAN
9872	RAUKE KALAN
8834	VARIO NANGAL
1274	KARIMPUR (LUDHIANA WEST)
1763	MOHI (LUDHIANA WEST)

Calculating accuracy: We use accuracy to gauge the performance of our prediction model. Here we use the Root Mean Square Error (RMSE) to gauge the performance of our system.

Table 2. Evaluation of Accuracy of the Recommendation

	RMSE Score:
Fold 1	0.9776
Fold 2	0.9789
Fold 3	0.9695
Fold 4	0.9810
Fold 5	0.9849
	Mean RMSE: 0.9784

Application User Interface (UI)

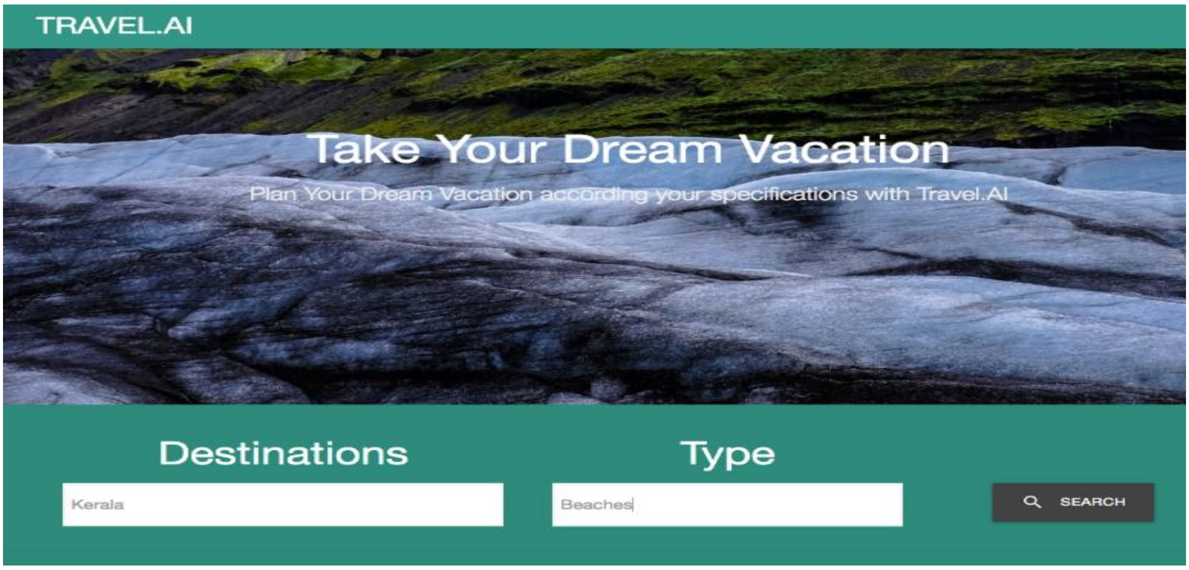

Figure 2. Application UI – User Preferences Screen

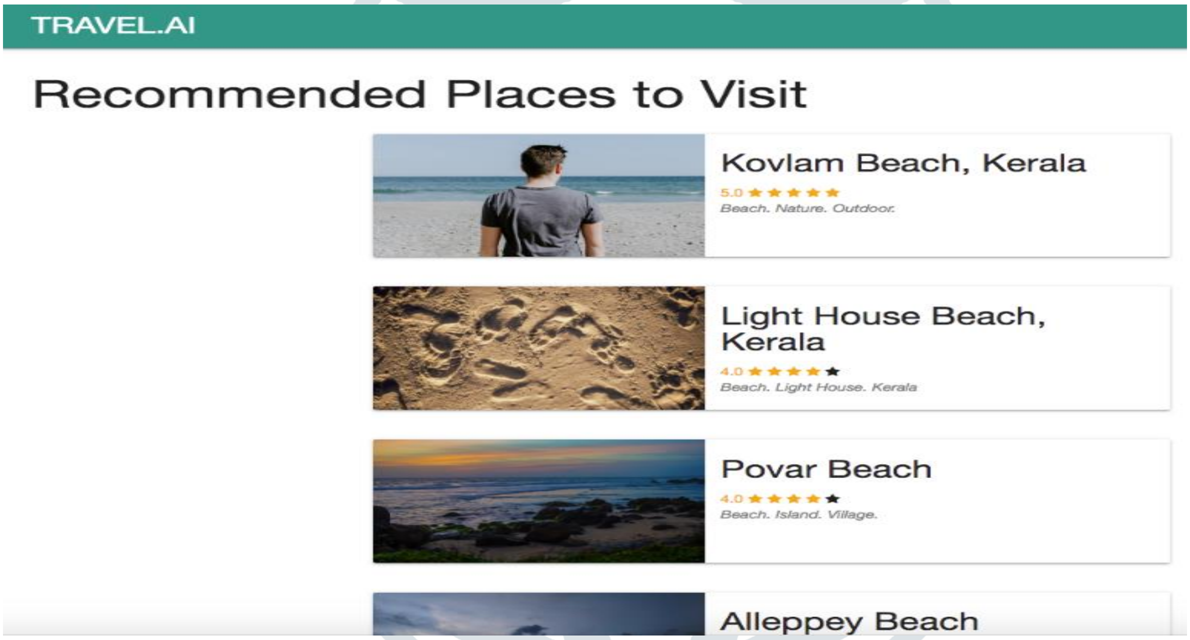

Figure 3. Application UI – Recommendation Results with ratings and metadata

V. CONCLUSION AND FUTURE SCOPE

In this paper we were able to experience the power of recommendation systems in the travel domain. Through recommendation systems the task of planning a trip to an unknown destination is become a much more efficient and easier task. Using hybrid recommendation system we were able to overcome the major cold start problem that comes with collaborative filtering algorithms. The hybrid recommendation system outperforms the earlier systems which used techniques such as tracking digital footprints, keyword-aware representative travel route framework, and case-based reasoning.

The collaborative filters used so far are known as memory-based filters. This is because they only make use of similarity based metrics to come up with their results. The system can further improve the relevance of recommendations by adding supervised and unsupervised machine learning techniques.

REFERENCES

[1] Z. Bahramian, R.A. Abbaspour, C. Claramunt, “A cold start context-aware recommender system for tour planning using artificial neural network and case based reasoning”, Mobile Information Systems, 2017.

[2] R. Banik, “Hands-On Recommendation Systems with Python: Start building powerful and personalized, recommendation engines with Python”, Packt Publishing Ltd, 2018.

[3] S. Kotiloglu, T. Lappas, K. Pelechrinis, P.P Repoussis, “Personalized multi-period tour recommendations”, Tourism Management 62, pp. 76-88, 2017.

[4] Z. Yu, H. Xu, Z. Yang, B. Gou, “Personalized travel package with multi-point-of-interest recommendation based on crowdsourced user footprints”, IEEE Transactions on Human-Machine Systems, Vol. 46, No. 1, pp. 151-158, 2016.

[5] Y.T. Wen, J. Yeo, W.C Peng, S.W. Hwang, “Efficient keyword-aware representative travel route recommendation”, IEEE Transactions on Knowledge and Data Engineering, Vol. 29, No. 8, pp. 1639-1652, 2017.

[6] M.J. Pazzani, D. Billsus, “The Adaptive Web”, Springer, Berlin, Heidelberg, pp. 325-341, 2007.

[7] A. Aizawa, “An information-theoretic perspective of tf–idf measures”, Information Processing & Management, Vol. 39(1), pp.45-65, 2003.

[8] S. Gong, “A collaborative filtering recommendation algorithm based on user clustering and item clustering”, JSW, Vol. 5(7), pp.745-752, 2010.

[9] G. Lekakos, P. Caravelas, “A hybrid approach for movie recommendation”, Multimedia tools and applications, Vol. 36(1-2), pp.55-70, 2008.

[10] E. Hallmark, "Points of Interest POI Database", Kaggle, 2018