A Cocktail Approach for Travel Package Recommendation

NALLAPU SHIRISHA¹, G. UMA²

¹PG Scholar, Sindhura College of Engineering and Technology, Godavari, TS, India, E-mail: nallapusrirsha10@gmail.com.
²Asst Prof, Sindhura College of Engineering and Technology, Godavari, TS, India, E-mail: umag660@gmail.com.

Abstract: Recent years have witnessed an increased interest in recommender systems. Despite significant progress in this field, there still remain numerous avenues to explore. Indeed, this paper provides a study of exploiting online travel information for personalized travel package recommendation. A critical challenge along this line is to address the unique characteristics of travel data, which distinguish travel packages from traditional items for recommendation. To that end, in this paper, we first analyze the characteristics of the existing travel packages and develop a tourist-area-season topic (TAST) model. This TAST model can represent travel packages and tourists by different topic distributions, where the topic extraction is conditioned on both the tourists and the intrinsic features (i.e., locations, travel seasons) of the landscapes. Then, based on this topic model representation, we propose a cocktail approach to generate the lists for personalized travel package recommendation. Furthermore, we extend the TAST model to the tourist-relation-area-season topic (TRAST) model for capturing the latent relationships among the tourists in each travel group. Finally, we evaluate the TAST model, the TRAST model, and the cocktail recommendation approach on the real-world travel package data. Experimental results show that the TAST model can effectively capture the unique characteristics of the travel data and the cocktail approach is, thus, much more effective than traditional recommendation techniques for travel package recommendation. Also, by considering tourist relationships, the TRAST model can be used as an effective assessment for travel group formation.

Keywords: Travel Package, Recommender Systems, Cocktail, Topic Modeling, Collaborative Filtering.

I. INTRODUCTION

A San emerging trend, more and more travel companies provide online services. However, the rapid growth of online travel information imposes an increasing challenge for tourists who have to choose from a large number of available travel packages for satisfying their personalized needs. Moreover, to increase the profit, the travel companies have to understand the preferences from different tourists and serve more attractive packages. Therefore, the demand for intelligent travel services is expected to increase dramatically. Since recommender systems have been successfully applied to enhance the quality of service in a number of fields, it is natural choice to provide travel package recommendations. Actually, recommendations for tourists have been studied before, and to the best of our knowledge, the first operative tourism recommender system was introduced by Delgado and Davidson. Despite of the increasing interests in this field, the problem of leveraging unique features to distinguish personalized travel package recommendations from traditional recommender systems remains pretty open. Indeed, there are many technical and domain challenges inherent in designing and implementing an effective recommender system for personalized travel package recommendation. First, travel data are much fewer and sparser than traditional items, such as movies for recommendation, because the costs for a travel are much more expensive than for watching a movie. Second, every travel package consists of many landscapes (places of interest and attractions), and, thus, has intrinsic complex spatio-temporal relationships. For example, a travel package only includes the landscapes which are geographically colocated together. Also, different travel packages are usually developed for different travel seasons.

Therefore, the landscapes in a travel package usually have spatial temporal autocorrelations. Third, traditional recommender systems usually rely on user explicit ratings. However, for travel data, the user ratings are usually not conveniently available. Finally, the traditional items for recommendation usually have a long period of stable value, while the values of travel packages can easily depreciate over time and a package usually only lasts for a certain period of time. The travel companies need to actively create new tour packages to replace the old ones based on the interests of the tourists. To address these challenges, in our preliminary work, we proposed a cocktail approach on personalized travel package recommendation. Specifically, we first analyze the key characteristics of the existing travel packages. Along this line, travel time and travel destinations are divided into different seasons and areas. Then, we develop a tourist-area-season topic (TAST) model, which can represent travel packages and tourists by different topic distributions. In the TAST model, the extraction of topics is conditioned on both the tourists and the intrinsic features (i.e., locations, travel seasons) of the landscapes. As a result, the TAST model can well represent the content of the travel packages and the interests of the tourists.
Based on this TAST model, a cocktail approach is developed for personalized travel package recommendation by considering some additional factors including the seasonal behaviors of tourists, the prices of travel packages, and the cold start problem of new packages. Finally, the experimental results on real-world travel data show that the TAST model can effectively capture the unique characteristics of travel data and the cocktail recommendation approach performs much better than traditional techniques. In this paper, we further study some related topic models of the TAST model, and explain the corresponding travel package recommendation strategies based on them. Also, we propose the tourist-relation-area-season topic (TRAST) model, which helps understand the reasons why tourists form a travel group. This goes beyond personalized package recommendations and is helpful for capturing the latent relationships among the tourists in each travel group. In addition, we conduct systematic experiments on the real-world data. These experiments not only demonstrate that the TRAST model can be used as an assessment for travel group automatic formation but also provide more insights into the TAST model and the cocktail recommendation approach. In summary, the contributions of the TAST model, the cocktail approaches, and the TRAST model for travel package recommendations are shown in Fig. 1, where each dashed rectangular box in the dashed circle identifies a travel group and the tourists in the same travel group are represented by the same icons.

![Fig. 1. An illustration of the paper contribution.](image)

**II. LITERATURE SURVEY**

A. Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions

This paper presents an overview of the field of recommender systems and describes the current generation of recommendation methods that are usually classified into the following three main categories: content-based, collaborative, and hybrid recommendation approaches. This paper also describes various limitations of current recommendation methods and discusses possible extensions that can improve recommendation capabilities and make recommender systems applicable to an even broader range of applications. These extensions include, among others, an improvement of understanding of users and items, incorporation of the contextual information into the recommendation process, support for multicriteria ratings, and a provision of more flexible and less intrusive types of recommendations. Recommender systems made significant progress over the last decade when numerous content-based, collaborative, and hybrid methods were proposed and several “industrial-strength” systems have been developed. However, despite all of these advances, the current generation of recommender systems surveyed in this paper still requires further improvements to make recommendation methods more effective in a broader range of applications. In this paper, we reviewed various limitations of the current recommendation methods and discussed possible extensions that can provide better recommendation capabilities. These extensions include, among others, the improved modeling of users and items, incorporation of the contextual information into the recommendation process, support for multcriteria ratings, and provision of a more flexible and less intrusive recommendation process. We hope that the issues presented in this paper will advance the discussion in the recommender systems community about the next generation of recommendation technologies.

B. Map-Based Interaction with a Conversational Mobile Recommender System

Recommender systems are information search and decision support tools used when there is an overwhelming set of options to consider or when the user lacks the domain-specific knowledge necessary to take autonomous decisions. They provide users with personalized recommendations adapted to their needs and preferences in a particular usage context. In this paper, we present an approach for integrating recommendation and electronic map technologies to build a map-based conversational mobile recommender system that can effectively and intuitively support users in finding their desired products and services. The results of our real-user study show that integrating map-based visualization and interaction in mobile recommender systems improves the system recommendation effectiveness and increases the user satisfaction.

In this paper we have presented an approach for integrating recommendation and electronic map technologies to build a map-based mobile recommender system that can effectively and intuitively provide personalized recommendations to mobile users. Our real-user study showed that the map-based interface is more effective than the list-based interface that is typically used in recommender systems. We also found that the integration of a map-based interface in a recommender system increases user satisfaction. There are several open issues that still must be studied. First, we did not investigate how different mappings of rank values to colors could
influence the user decision. It would be interesting to analyze to what extent we can push the concept of “green” (i.e., strongly recommended) item, and how an over abundance of “green” items can influence the user decision. Second, in MapMobyRek the user cannot review the recommendation lists produced in previous cycles. It could be helpful if the system records the recommendation states and supports the user (with an “undo” button) to review a previous recommendation state. Another important topic to investigate is how to integrate recommendations on different product types (e.g., a restaurant and an itinerary) and suggest them as a “package”. This would be very important in a real commercial exploitation of the system since typically services that could generate profit are advertised in conjunction with points of interests that the tourist may like to visit.

C. Hybrid Web Recommender Systems
Adaptive web sites may offer automated recommendations generated through any number of well-studied techniques including collaborative, content-based and knowledge-based recommendation. Each of these techniques has its own strengths and weaknesses. In search of better performance, researchers have combined recommendation techniques to build hybrid recommender systems. This chapter surveys the space of two-part hybrid recommender systems, comparing four different recommendation techniques and seven different hybridization strategies. Implementations of 41 hybrids including some novel combinations are examined and compared. The study finds that cascade and augmented hybrids work well, especially when combining two components of differing strengths. This chapter has more fully characterized each of 53 hybrid types and described experiments that compare the performance of a subset of the design space. The experiments cover the space of possible hybrid recommender systems available with four basic recommendation algorithms: content-based, standard collaborative, heuristic collaborative and knowledge-based. Six types of combinations were explored: weighted, switching, feature combination, feature augmentation, cascade and meta-level, for a total of 41 different systems. Due to data and methodological limitations, demographic recommendation and mixed hybrids were not explored. Because two different collaborative algorithms were explored, the 41 systems evaluated represent 24 of the 53 spaces in this table, including 12 recommenders with no previous known examples. Of course, any such study is by its nature limited by the peculiarities of the data and the recommendation domain. The Entree data set is relatively small (just over ¼ million ratings), the profiles are short and the ratings are implicit and heavily skewed to the negative. It would be valuable to repeat this study in a different recommendation domain with different products and a set of user profiles with different characteristics. In particular, it is unfortunate that the circumstances of this study allow only very limited findings with respect to meta-level recommendation.

III. ANALYSIS
The Systems Development Life Cycle (SDLC), or Software Development Life Cycle in systems engineering, information systems and software engineering, is the process of creating or altering systems, and the models and methodologies that people use to develop these systems. In software engineering the SDLC concept underpins many kinds of software development methodologies. These methodologies form the framework for planning and controlling the creation of an information system the software development process.

A. Existing System
There are many technical and domain challenges inherent in designing and implementing an effective recommender system for personalized travel package recommendation.

1. Travel data are much fewer and sparser than traditional items, such as movies for recommendation, because the costs for a travel are much more expensive than for watching a movie.

2. Every travel package consists of many landscapes (places of interest and attractions), and, thus, has intrinsic complex spatio-temporal relationships. For example, a travel package only includes the landscapes which are geographically colocated together. Also, different travel packages are usually developed for different travel seasons. Therefore, the landscapes in a travel package usually have spatial temporal autocorrelations.

3. Traditional recommender systems usually rely on user explicit ratings. However, for travel data, the user ratings are usually not conveniently available.

B. Proposed System
In this paper, we aim to make personalized travel package recommendations for the tourists. Thus, the users are the tourists and the items are the existing packages, and we exploit a real-world travel data set provided by a travels for building recommender systems. we develop a tourist-area-season topic (TAST) model, which can represent travel packages and tourists by different topic distributions. In the TAST model, the extraction of topics is conditioned on both the tourists and the intrinsic features (i.e., locations, travel seasons) of the landscapes. Based on this TAST model, a cocktail approach is developed for personalized travel package recommendation by considering some additional factors including the seasonal behaviors of tourists, the prices of travel packages, and the cold start problem of new packages.

Advantages of Proposed System:
- Represent the content of the travel packages and the interests of the tourists.
- TAST model can effectively capture the unique characteristics of travel data.
- The cocktail recommendation approach performs much better than traditional techniques.

C. Process Model Used With Justification
SDLC (Umbrella Model):
SDLC is nothing but Software Development Life Cycle. It is a standard which is used by software industry to develop good software.

**Class diagram:** The class diagram is the main building block of object-oriented modeling. It is used both for general conceptual modeling of the systematic of the application, and for detailed modeling translating the models into programming code. Class diagrams can also be used for data modeling. The classes in a class diagram represent both the main objects, interactions in the application and the classes to be programmed. In the diagram, classes are represented with boxes which contain three parts:

- The upper part holds the name of the class
- The middle part contains the attributes of the class
- The bottom part gives the methods or operations the class can take or undertake

**IV. TESTING**

**A. Implementation and Testing**

Implementation is one of the most important tasks in project is the phase in which one has to be cautious because all the efforts undertaken during the project will be very interactive. Implementation is the most crucial stage in achieving successful system and giving the users confidence that the new system is workable and effective. Each program is tested individually at the time of development using the sample data and has verified that these programs link together in the way specified in the program specification. The computer system and its environment are tested to the satisfaction of the user.

**B. Implementation**

The implementation phase is less creative than system design. It is primarily concerned with user training, and file conversion. The system may be requiring extensive user training. The initial parameters of the system should be modified as a result of a programming. A simple operating procedure is provided so that the user can understand the different functions clearly and quickly. The different reports can be obtained either on the inkjet or dot matrix printer, which is available at the disposal of the user. The proposed system is very easy to implement. In general implementation is used to mean the process of converting a new or revised system design into an operational one.

**C. Testing**

Testing is the process where the test data is prepared and is used for testing the modules individually and later the validation given for the fields. Then the system testing takes place which makes sure that all components of the system property functions as a unit. The test data should be chosen such that it passed through all possible condition. Actually testing is the state of implementation which aimed at ensuring that the system works accurately and efficiently before the actual operation commence. The following is the description of the testing strategies, which were carried out during the testing period.

1. **System Testing**

   Testing has become an integral part of any system or project especially in the field of information technology. The importance of testing is a method of justifying, if one is ready to move further, be it to be check if one is capable to withstand the rigors of a particular situation cannot be underplayed and that is why testing before development is so critical. When the software is developed before it is given to user the software must be tested whether it is solving the purpose for which it is developed. This testing involves various types through which one can ensure the software is reliable. The program was tested logically and pattern of execution of the program for a set of data are repeated. Thus the code was exhaustively checked for all possible correct data and the outcomes were also checked.

2. **Module Testing**

   To locate errors, each module is tested individually. This enables us to detect error and correct it without affecting any other modules. Whenever the program is not satisfying the required function, it must be corrected to get the required result. Thus all the modules are individually tested from bottom up starting with the smallest and lowest modules and proceeding to the next level. Each module in the system is
A Cocktail Approach for Travel Package Recommendation

tested separately. For example the job classification module is tested separately. This module is tested with different job and its approximate execution time and the result of the test is compared with the results that are prepared manually. The comparison shows that the results proposed system works efficiently than the existing system. Each module in the system is tested separately. In this system the resource classification and job scheduling modules are tested separately and their corresponding results are obtained which reduces the process waiting time.

3. Integration Testing
After the module testing, the integration testing is applied. When linking the modules there may be chance for errors to occur, these errors are corrected by using this testing. In this system all modules are connected and tested. The testing results are very correct. Thus the mapping of jobs with resources is done correctly by the system.

4. Acceptance Testing
When that user fined no major problems with its accuracy, the system passes through a final acceptance test. This test confirms that the system needs the original goals, objectives and requirements established during analysis without actual execution which elimination wastage of time and money acceptance tests on the shoulders of users and management, it is finally acceptable and ready for the operation.

V. RESULTS

Home screen:

Fig3. Click on Tour Company to login as the company admin.

Fig4. Tour company home screen after successful login.

Fig5. Click on load tour package to load the different packages.

Registration screen:

Fig6. Register as a new user.

Fig7. Login as the registered user.

Fig8.
VI. CONCLUSION

In this paper, we present study on personalized travel package recommendation. Specifically, we first analyzed the unique characteristics of travel packages and developed the TAST model, a Bayesian network for travel package and tourist representation. The TAST model can discover the interests of the tourists and extract the spatial-temporal correlations among landscapes. Then, we exploited the TAST model for developing a cocktail approach on personalized travel package recommendation. This cocktail approach follows a hybrid recommendation strategy and has the ability to combine several constraints existing in the real-world scenario. Furthermore, we extended the TAST model to the TRAST model, which can capture the relationships among tourists in each travel group. Finally, an empirical study was conducted on real-world travel data. Experimental results demonstrate that the TAST model can capture the unique characteristics of the travel packages, the cocktail approach can lead to better performances of travel package recommendation, and the TRAST model can be used as an effective assessment for travel group automatic formation. We hope these encouraging results could lead to many future work.

VII. REFERENCES