© 2015 IJSRSET | Volume 1 | Issue 2 | Print ISSN : 2395-1990 | Online ISSN : 2394-4099
Themed Section: Engineering and Technology

A Ranking Based Combined Approach for Recommending Travel Packages

N. R. Rejin Paul, G. Karpagam, R. Sindhuja, S. Yamini
Dept of Computer Science and Engineering, Velammal Institute of Technology, Anna University, Chennai, Tamilnadu, India

ABSTRACT

The 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. To that end, in this system, there is a need to 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: TAST, TRAST, Cocktail, Ranking

I. INTRODUCTION

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 collocated 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, this project 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, this project develops 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 project, we further study some related topic models of the TAST model, and explain the corresponding travel package recommendation strategies based on them. Also, this project proposes 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, 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.

II. METHODS AND MATERIAL

A. Related Works

According to **Gregory D. Abowd, Christopher G. Atkeson, Jason Hong, Sue Long; Rob Kooper and Mike Pinkerton**, this project describe the concept based on the technique of Cyberguide prototypes. Future computing environments promise to free the user from the constraints of stationary desktop computing, yet relatively few researchers are investigating what applications maximally benefit from mobility. We describe the architecture and features of a variety of Cyberguide prototypes developed for indoor and outdoor use on a number of different hand-held platforms.

The long-term goal is an application that knows where the tourist is, what she is looking at, can predict and answer questions she might pose, and provide the ability to interact with other people and the environment. Our short-term goal was to prototype versions of Cyberguide on commercially available PDAs and pen-based PCs in which context-awareness simply meant the current physical position and orientation of the Cyberguide unit (and since it is hand-held, this locates the user as well). Position information improves the utility of a tour guide application.

According to **Gediminas Adomavicius and Alexander Tuzhilin**, the content based recommendation represents that the user is recommended items similar to the ones the user preferred in the past; The Collaborative recommendations mentions that the user is recommended items that people with similar tastes and preferences liked in the past; The Hybrid approaches mentions that these methods combine collaborative and content-based methods.

Although the roots of recommender systems can be traced back to the extensive work in the cognitive science, approximation theory, information retrieval, forecasting theories, and also have links to management science, and also to the consumer choice modeling in marketing, recommender systems emerged as an independent research area in the mid-1990's when researchers started focusing on recommendation problems that *explicitly* rely on the ratings structure. In its most common formulation, the recommendation problem is reduced to the problem of estimating *ratings* for the items that have not been seen by a user. Once we can estimate ratings for the yet unrated items, we can recommend to the user the item(s) with the highest estimated rating(s).

According **to Deepak Agarwal and Bee-Chung Chen**, this project uses the technique of Gaussian linear regression and Latent Dirichlet Allocation (LDA) priors respectively. We show our model is accurate, interpretable and handles both cold-start and warm-start scenarios seamlessly through a single model.

The key idea of our method is to let the user factors (or profiles) take values in an Euclidean space as in existing factorization models, but assign item factors through a richer prior based on Latent Dirichlet Allocation (LDA). The main idea in LDA is to attach a *discrete* latent factor to each word of an item that can take K different values (K topics) and produce item topics by averaging the per-word topics in the item. The key is the ability of fLDA to learn these scores automatically from the data. The simultaneous estimation of both user profiles and topic attribution makes our method distinct from recent work called sLDA that incorporates a response variable (like reviews on articles) in deciding LDA topics through a *global* regression (while our model performs per-user local regression). In fact, if we assume all users share the same profile, fLDA reduces to sLDA.

The topic representation of items in fLDA also provide interpretability and may help in explaining recommendations to users in applications. The LDA model is well known to provide such interpretation since the probability mass of topics tend to be concentrated on a small set of words. This interpretability is important for a number of reasons.

According to Olga **Averjanova, Francesco Ricci, and Quang Nhat Nguyen**, the map based techniques are followed in this project. The map-based interface of MapMobyRek focused on the following user functions: The first function is to enter the search query by specifying preferences for item features. Next is to see the system's recommendations on the map. The other function is to recognize immediately the differences between good and weak recommendations. Next is to compare two selected recommendations. To input critiques to the recommended items and to see on the map how the expressed critique influences the system's recommendations, and finally to select the best items.
Dynamic Tour Guide (DTG) is a mobile tour guide system that helps travelers in discovering a destination.
After making a critique, it might be difficult for the user to get an immediate feedback of 1) how the expressed critique influences (changes) the ranking of the recommended items, or 2) the appearance of new recommendations, or 3) the disappearance of previously recommended items. To determine which items lose/gain higher/lower recommendation score, and
consequently their rank, in MobyRek the user has to remember the previous result set.

According to **David M. Blei, Andrew Y. Ng and Michael I. Jordan,** this project describes *latent Dirichlet allocation* (LDA), a generative probabilistic model for collections of discrete data such as text corpora. LDA is a three-level hierarchical Bayesian model, in which each item of a collection is modeled as a finite mixture over an underlying set of topics.

In the context of text modeling, the topic probabilities provide an explicit representation of a document. We present efficient approximate inference techniques based on variational methods and an EM algorithm for empirical Bayes parameter estimation. We report results in document modeling, text classification, and collaborative filtering, comparing to a mixture of unigrams model and the probabilistic LSI model.

A significant step forward in this regard was made by Hofmann (1999), who presented the *probabilistic LSI (pLSI)* model, also known as the *aspect model*, as an alternative to LSI. The pLSI approach, models each word in a document as a sample from a mixture model, where the mixture components are multinomial random variables that can be viewed as representations of "topics." Each document is represented as a list of mixing proportions for these mixture components and thereby reduced to a probability distribution on a fixed set of topics. This distribution is the "reduced description" associated with the document. In pLSI, each document is represented as a list of numbers , and there is no generative probabilistic model for these numbers. This leads to several problems: (1) the number of parameters in the model grows linearly with the size of the corpus, which leads to serious problems with overfitting, and (2) it is not clear how to assign probability to a document outside of the training set.

B. Proposed System

In the proposed system, to make the travel selection easier, the recommendation is provided to the tourists. This project first analyzes the characteristics of the existing travel packages and develops 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, this project evaluates the TAST model, the TRAST model, and the cocktail recommendation approach on the real-world travel package data.

To address these challenges, in our preliminary work, this project 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 project, 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.

III. RESULTS AND DISCUSSION

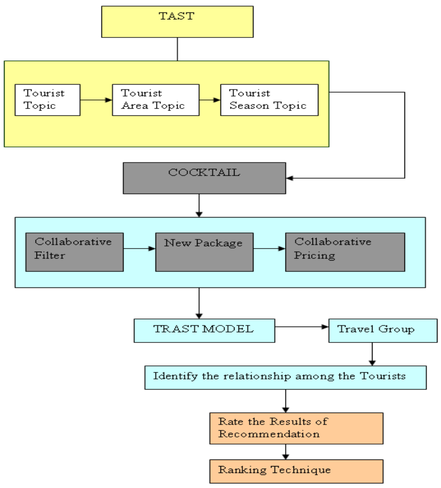


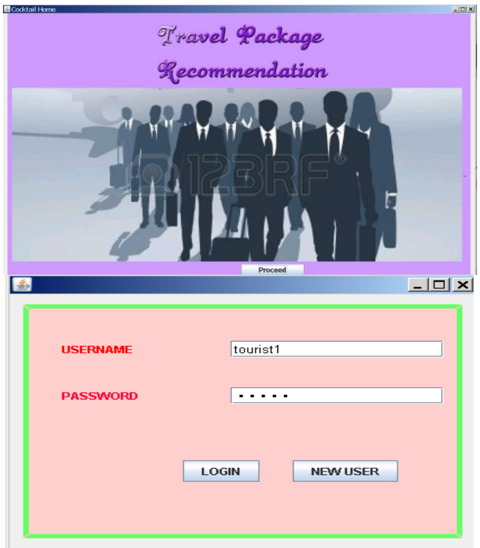
Figure 1: System Architecture

List of Modules

i. USER MODULE

This module creates the users required in this project. The user uses the system to get the travel package recommendation. In this module, the input provided by the user is processed in order to validate and provide the appropriate recommendation to the user.

This module helps to login or register into the system. If the user is new, then the user has to register themselves .The users who have already registered can login to the system in order to use it and get recommendations. The user uses the system to get the travel package recommendation.

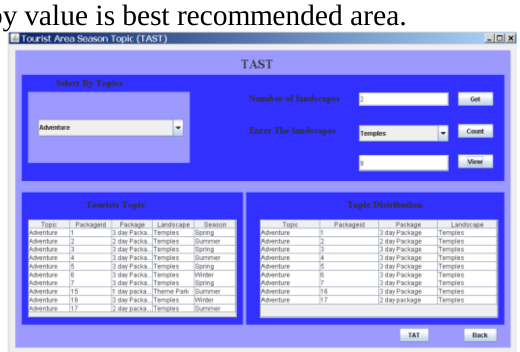


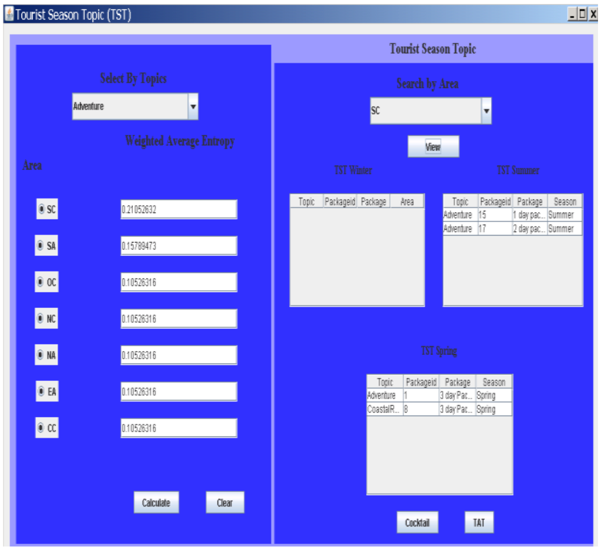
ii. TAST MODULE

The Tourist Area Season Topic Model is composed of the following models.

- TT Model
- TAT Model
- TST Model

In this model Weighted Area Entropy (WAE) is calculated as the ratio of no of visits to that area to the total number of data sets. The area which has highest entropy value is best recommended area.

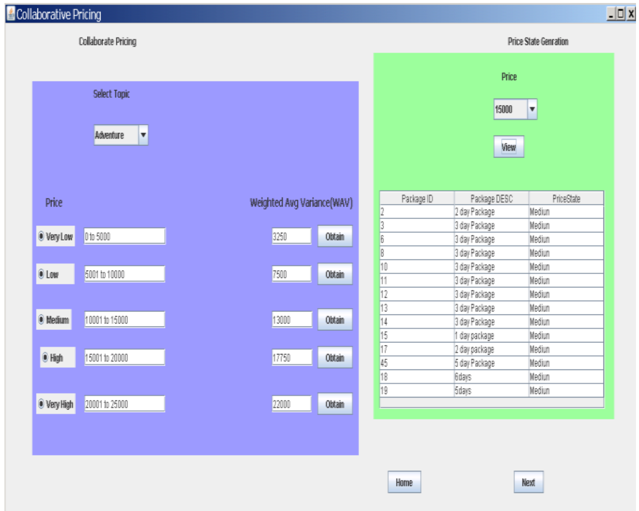




iii. COCKTAIL MODULE

The cocktail module is a hybrid recommendation strategy model. Collaborative filtering will be used for ranking the candidate packages. The new packages are added into the candidate list by computing similarity with the candidate packages generated previously.

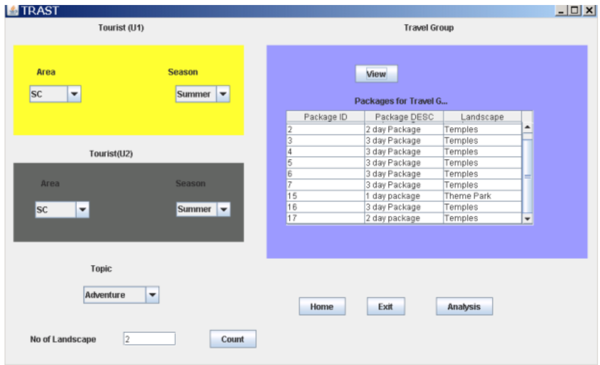
Collaborative filtering systems have many forms, but many common systems can be reduced to two steps: Look for users who share the same rating patterns with the active user (the user whom the prediction is for).Use the ratings from those like-minded users found to calculate a prediction for the active user. The neighborhood-based algorithm calculates the similarity between two users or items, produces a prediction for the user taking the weighted average of all the ratings. Similarity computation between items or users is an important part of this approach. Multiple mechanisms such as Pearson correlation and vector cosine based similarity are used for this. Collaborative pricing is used to predict the possible price distribution of each tourist and reorder the packages.



iv. TRAST MODULE

The TRAST module identifies the latent relationship between two tourists for each package in the dataset. By considering tourist relationships, the TRAST model can be used as an effective assessment for travel group formation.

The results of these models are being clustered in order to provide a valid recommendation to the tourists.



v. RECOMMENDATION MODULE

Based on the clustering process using the kmeans algorithm, the results are clustered and processed according to the need of the user based on the topic, area and season respectively. Moreover, the recommendation is also based on the pricing of the previously travelled tourists.

vi. RANKING MODULE

In the analysis module, the performance values of the three models are shown. Based on the results produced by the analysis, this project can conclude the model that works efficiently in order to provide better recommendation to the users than the other models.The ranking module is implemented by means of a selection ranking algorithm. The selection algorithm is used for finding the kth smallest number in a list or array; such a number is called the kth order statistic. This includes the cases of finding the minimum, maximum, and median elements.

vii. TECHNIQUES AND ALGORITHMS

K-means algorithm

K-means algorithm criterion function adopts square error criterion, be defined as:

$$E= \sum_{j=1}^{k} \sum_{i=1}^{n} \left\| x_{i-} m_j \right\|^2$$

In which, E is total square error of all the objects in the data cluster, *xi* bellows to data object set, *mi* is mean value of cluster *Ci* (x and m are both multi-dimensional). The function of this criterion is to make the generated cluster be as compacted and independent as possible. The distance between data points and the cluster center is identified. The distance formula of data point xi and cluster center kj defined as following

$$d_{j,i}= \sqrt{(x_{i1}-k_{j1})^2+(x_{i2}-k_{j2})^2+\cdots......+(x_{iw}-k_{j1w})^2}$$

Where w represents the number of attributes of the data points xi.

The clustering algorithm adds the weight of data point to the cluster center. Data points near the center of the cluster weights, on the contrary, the value of data points away from the cluster center is less weight. The formula of cluster center defined as follows:

$$k= \frac{d_{j_h}}{D}\, x_{j1}+\frac{d_{j(h-1)}}{D}\, x_{j2}+......+\frac{d_{j2}}{D}\, x_{j_{(h-2)}} + \frac{d_{j_1}}{D}\, x_{j_h}$$

where j represents the jth cluster, h is the number of data points in the cluster, djh represents the distance between the hth data point which belongs to cluster c and cluster center. And with the restriction of

$$d_{j1} \leq d_{j2} \leq \cdots \leq d_{j_h}, \frac{d_{j1}}{D} + \frac{d_{j2}}{D} + \ldots \frac{d_{j_h}}{D} = 1$$

The Euclidean distance between data points and the cluster center is identified. The distance between data point and the cluster center determine the cluster which data point belongs to, the formula of Euclidean distance is defined as follows:

$$d_{j_1}=\left(1-\frac{\sigma_j}{\sigma}\right) d_{j_1}$$

where j represents the jth cluster cj , i represents the ith data point xi, dji is the Euclidean distance between data point xi and the cluster center cj , σi represents the squares error of the cluster cj , σ is the squares error sum of the K clusters c.

Selection Ranking algorithm

Selection Rank is a well-known algorithm in computer science to find the ith smallest or largest element in an array in expected linear time.

The simplest case of a selection algorithm is finding the minimum (or maximum) element by iterating through the list, keeping track of the running minimum, the minimum so far or maximum and can be seen as related to the selection sort. Conversely, the hardest case of a selection algorithm is finding the median, and this necessarily takes *n*/2 storage. In fact, a specialized median-selection algorithm can be used to build a general selection algorithm, as in median of medians. The best-known selection algorithm is quick select, which is related to quick sort; like quick sort, it has (asymptotically) optimal average performance, but poor worst-case performance, though it can be modified to give optimal worst-case performance as well.

The basic algorithm for finding the ith smallest elements goes like this:

* Pick a random element in the array and use it as a ‘pivot’. Move all elements smaller than that element to one side of the array, and all elements larger to the other side.

*If there are exactly i elements on the right, then you just find the smallest element on that side.

Otherwise, if the right side is bigger than i, repeat the algorithm on the right. *If the right side is smaller than i, repeat the algorithm on the left for i – right.size().

Given this algorithm,you can either:
*Tweak it to use the existing partitions to find the largest I elements.

*Or, once you find the ith largest element, run through the array again to return all elements greater than or equal. This algorithm has expected O(n) time.

Pseudocode for Selection Ranking

function select(list[1..n], k)
for i from 1 to k
minIndex = i minValue = list[i]

```
for j from i+1 to n
if list[j] < minValue
        minIndex = j
        minValue = list[j]
swap list[i] and list[minIndex]
return list[k]
```

IV. FUTURE ENHANCEMENT

In the future, this project can be extended to a distributed scenario in which the project can be hosted and can be used by the tourists online. During this online utilization of this system, the travel package recommendation will be provided to the users online so that the tourists will be able to book their travel package on the spot.

V. CONCLUSION

In this work, a study on personalized travel package recommendation is presented. Specifically, this project 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, this project exploited the TAST model for developing a cocktail approach on personalized travel package recommendation. This cocktail approach follows a hybrid recommendation strategy and as the ability to combine several constraints existing in the real-world scenario. Furthermore, this project extended the TAST model to the TRAST model, which can capture the relationships among tourists in each travel group. Finally, an empirical study this project 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.

VI. REFERENCES

[1] G.D. Abowd et al., "Cyber-Guide: A Mobile Context-Aware Tour Guide," Wireless Networks, vol. 3, no. 5, pp. 421-433, 1997.

[2] G. Adomavicius and A. Tuzhilin, "Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions," IEEE Trans. Knowledge and Data Eng., vol. 17, no. 6, pp. 734-749, June 2005.

[3] D. Agarwal and B. Chen, "fLDA: Matrix Factorization through Latent Dirichlet Allocation," Proc. Third ACM Int’l Conf. Web Search and Data Mining (WSDM ’10), pp. 91-100, 2010.

[4] O. Averjanova, F. Ricci, and Q.N. Nguyen, "Map-Based Interaction with a Conversational Mobile Recommender System," Proc. Second Int’l Conf. Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM ’08), pp. 212-218, 2008.

[5] D.M. Blei, Y.N. Andrew, and I.J. Michael, "Latent Dirichlet Allocation," J. Machine Learning Research, vol. 3, pp. 993-1022, 2003.

[6] R. Burke, "Hybrid Web Recommender Systems," The Adaptive Web, vol. 4321, pp. 377-408, 2007.

[7] B.D. Carolis, N. Novielli, V.L. Plantamura, and E. Gentile, "Generating Comparative Descriptions of Places of Interest in the Tourism Domain," Proc. Third ACM Conf. Recommender Systems (RecSys ’09), pp. 277-280, 2009.

[8] F. Cena et al., "Integrating Heterogeneous Adaptation Techniques to Build a Flexible and Usable Mobile Tourist Guide," AI Comm., vol. 19, no. 4, pp. 369-384, 2006.

[9] W. Chen, J.C. Chu, J. Luan, H. Bai, Y. Wang, and E.Y. Chang, "Collaborative Filtering for Orkut Communities: Discovery of User Latent Behavior," Proc. ACM 18th Int’l Conf. World Wide Web (WWW ’09), pp. 681-690, 2009.

[10] N.A.C. Cressie, Statistics for Spatial Data. Wiley and Sons, 1991.

[11] J. Delgado and R. Davidson, "Knowledge Bases and User Profiling in Travel and Hospitality Recommender Systems" Proc. ENTER 2002 Conf. (ENTER ’02), pp. 1-16, 2002.

[12] U.M. Fayyad and K.B. Irani, "Multi-Interval Discretization of Continuous-Valued Attributes for Classification Learning," Proc. Int’l Joint Conf. Artificial Intelligence (IJCAI), pp. 1022-1027, 1993.

[13] F. Fouss et al., "Random-Walk Computation of Similarities between Nodes of a Graph with Application to Collaborative Recommendation," IEEE Trans. Knowledge and Data Eng., vol. 19, no. 3, pp. 355-369, Mar. 2007.

[14] Y. Ge et al., "Cost-Aware Travel Tour Recommendation," Proc. 17th ACM SIGKDD Int’l Conf. Knowledge Discovery and Data Mining (SIGKDD ’11), pp. 983-991, 2011.

[15] Y. Ge et al., "An Energy-Efficient Mobile Recommender System," Proc. 16th ACM SIGKDD Int’l Conf. Knowledge Discovery and Data Mining (SIGKDD ’10), pp. 899-908, 2010.

[16] M. Gori and A. Pucci, "ItemRank: A Random-Walk Based Scoring Algorithm for Recommender Engines," Proc. 20th Int’l Joint Conf. Artificial Intelligence (IJCAI ’07), pp. 2766-2771, 2007.

[17] U. Gretzel, "Intelligent Systems in Tourism: A Social Science Perspective," Annals of Tourism Research, vol. 38, no. 3, pp. 757-779, 2011.

[18] T.L. Griffiths and M. Steyvers, "Finding Scientific Topics," Proc. Nat’l Academy of Sciences USA, vol. 101, pp. 5228-5235, 2004.

[19] Q. Hao et al., "Equip Tourists with Knowledge Mined from Travelogues," Proc. 19th Int’l Conf. World Wide Web (WWW ’10), pp. 401-410, 2010.

[20] J. Herlocker, J. Konstan, L. Terveen, and J. Riedl, "Evaluating Collaborative Filtering Recommender Systems," ACM Trans. Information Systems, vol. 22, no. 1, pp. 5-53, 2004.