Object-oriented Travel Package Recommendation

CHANG TAN, University of Science and Technology of China
QI LIU, University of Science and Technology of China
ENHONG CHEN, University of Science and Technology of China
HUI XIONG, Rutgers University
XIANG WU, University of Science and Technology of China

Providing better travel services for tourists is one of the important applications in urban computing. Though many recommender systems have been developed for enhancing the quality of travel service, most of them lack a systematic and open framework to dynamically incorporate multiple types of additional context information existing in the tourism domain, such as the travel area, season, and price of the travel packages. To that end, in this paper, we propose an open framework, *Objected-oriented Recommender System* (ORS), for the developers performing personalized travel package recommendation to the tourists. This framework has the ability to import all the available additional context information to the travel package recommendation process in a cost-effective way. Specifically, the different types of additional information are extracted and uniformly represented as feature-value pairs. Then, we define the Object, which is the collection of the feature-value pairs. We propose two models which can be used in the ORS framework for extracting the implicit relationships among Objects. *Objected-oriented Topic Model* (OTM) can extract the topics conditioned on the intrinsic feature-value pairs of the Objects. *Objected-oriented Bayesian Network* (OBN) can effectively infer the co-travel probability of two tourists by calculating the co-occurrence time of feature-value pairs belonging to different kinds of Objects. Based on the relationships mined by OTM or OBN, the recommendation list is generated by the collaborative filtering method. Finally, we evaluate these two models and the ORS framework on real-world travel package data, and the experimental results show that the ORS framework is more flexible in terms of incorporating additional context information, and thus leads to better performances for travel package recommendation. Meanwhile, for feature selection in ORS, we define the feature information entropy, and the experimental results demonstrate that using features with lower entropies usually lead to better recommendation results.

Categories and Subject Descriptors: H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Information Filtering*; H.2 [**Database Management**]: Database Application—*Data Mining*

General Terms: Algorithms, Experimentation

Additional Key Words and Phrases: Bayesian, Collaborative filtering, Object-oriented, Topic model, Travel

ACM Reference Format:
Tan, C., Liu, Q., Chen, E., Xiong, H., and Wu, X. 2013. Object-oriented Travel Package Recommendation. ACM Trans. Intell. Syst. Technol. 0, 0, Article 0 (2013), 26 pages.
DOI = 10.1145/0000000.0000000 http://doi.acm.org/10.1145/0000000.0000000

1. INTRODUCTION

With the rapid growth of travel industry, the tourism is becoming one of the key elements affecting urban development, and the need of fast and intelligent travel services has increased strongly during the last decades [Ricci 2002]. A major effort along this

Author's addresses: C. Tan, Q. Liu, E. Chen and X. Wu, School of Computer Science and Technology, University of Science and Technology of China; H. Xiong, Rutgers Business School, Rutgers University.

DOI 10.1145/0000000.0000000 http://doi.acm.org/10.1145/0000000.0000000

line is the development of travel recommender systems, which are significantly different from the classical recommender systems due to the specific characteristics of travel data and the recommendation objectives. For instance, given a large collection of historical travel data, travel package recommendation has a goal of recommending the suitable travel packages to the right tourists. Unlike traditional items (i.e. movies and books) for recommendation, travel packages usually include a set of selected landscapes and have a lot of additional context information, such as area and season, price, travel time and route constraints.

These additional information have significant impact on the choices of tourists. Taking the price and the time cost for movies and travel packages as an example, movies commonly have two-hours length and the similar prices, but travel packages can vary from one-day excursion to half-month luxury holiday. Thus, tourists must make a decision considering their funds and spare time. Therefore, there is an interactive process between the features of travel packages and the specific situation of the tourists. Actually, people have studied how to exploit some of these additional context information for enhancing travel package recommendation. For instance, [Ge et al. 2011] considered to incorporate the finance and time cost for travel package recommendation, and [Liu et al. 2011] tried to capture the correlations among two features (i.e., area and season) of the travel packages. However, both of them only took some specific features into consideration, and they lack the capability to exploit all the additional context information. Recently, the urban environment presents a new challenge about using multiple context data to improve travel recommendation. For instance, [Zheng et al. 2012] developed an integrated and effective mobile recommendation system including three algorithms to answer location-related queries for location-based services. Meanwhile, beyond the tourism domain, there are some recommendation works which exploit additional user/item features to improve recommendation results. For instance, for academic collaboration recommendation, [Tang et al. 2012] proposed the Cross-domain Topic Learning (CTL) model to highlight the existing relationships of authors through implicit topic layers and publications. However, these algorithms are not suitable for travel package recommendation, and more importantly, they do not pay close attention to the intrinsic connections among the features of users or items.

In summary, to the best of our knowledge, the existing studies usually consider additional context information in a case-by-case manner, and there is no systematic solution to simultaneously and dynamically incorporate multiple types of contexts. This motivates us to find novel methods for improving the recommendation effects.

1.1. Contributions

In this paper, we define a systematic solution for dealing with the multifarious context information. In this way, the extra overhead for processing different types of additional information will be avoided and thus more efficient recommendation methods can be proposed. Specifically, we are inspired by the idea of Object-oriented programming where the key-value pairs are used for saving information of Objects. Similarly, the users or items in recommender systems are also abstract concepts, it is natural to consider them as Objects following the Object-oriented programming. To that end, we propose to develop an open framework *Object-oriented Recommender System* (ORS) for developers, which has the ability to import all the available additional context information in the recommendation process in a systematic and cost-effective way.

Along the line of the development of the ORS framework, we first analyze the key characteristics of the travel packages and provide a new way to represent the travel data. Specifically, the different types of context information in the travel packages are extracted and represented as feature-value pairs and the **Object** is defined as the collection of these feature-value pairs. In the ORS framework, a travel record is an Object and thus can be represented by a collection of feature-value pairs. For instance, for a travel record, Alice (a 20-years-old girl) took a Hong Kong one-day tour in Summer 2011, so this travel record can be represented as {Name: Alice, Age: 20, Gender: female, Days: 1, Area: Hong Kong, Season: Summer, Year: 2011}. Similarly, we can also regard the tourists and the travel packages as Objects, for instance, a tourist owns a set of features about himself and his travel histories.

Then, we propose two models which can be used in the ORS framework to mine the implicit relationships (similarities) among the Objects. The first one is a novel topic model named *Objected-oriented Topic Model* (OTM) which considers the tourist correlations as the latent topics hidden in the collection of intrinsic feature-value pairs of the Objects. The second one is another simple Bayesian network model, *Objected-oriented Bayesian Network* (OBN), which can more efficiently infer the co-travel probabilities of two tourists by calculating the co-occurrence times of the feature-value pairs. Next, based on the relationships mined by OTM or OBN, the nearest neighbors for each tourist can be found and the recommendation list is generated by the collaborative filtering method. Finally, the ORS framework is completed for travel package recommendation by considering some additional factors including the annual behaviors of tourists as well as the cold start problem of new packages.

We evaluate these two similarity models (OTM and OBN) and the ORS framework on real-world travel package data, and the experimental results show that the ORS framework is more flexible in terms of incorporating additional context information, and thus performs much better for travel package recommendation than state-of-the-art recommendation methods. Meanwhile, for feature selection in ORS, we define and compute the feature information entropy using the OTM model, and the corresponding experimental results on the ORS framework demonstrate that using features with lower entropies usually lead to better recommendation results.

1.2. Outline

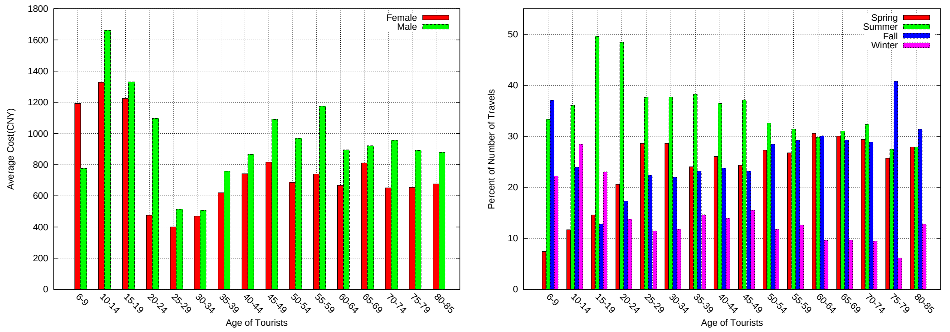
The rest of this paper is organized as follows. Section 2 introduces the travel data analysis and the basic concepts. Section 3 and Section 4 describe the details of the OTM model and the OBN model, respectively. In Section 5, we present the ORS framework for real-world applications based on OTM or OBN model. Experimental results are shown in Section 6. Section 7 discusses how to select useful features. After introducing some related research works in Section 8, we conclude the paper in Section 9.

2. CONCEPTS AND PRELIMINARIES

In this section, we first give the travel package recommendation scenario. Specifically, we aim to make personalized travel package (item) recommendation for the tourists (users). Then, we analyze the unique characteristics of the travel package data in detail and describe the correlations among the additional context information. By considering the different types of additional information uniformly represented as feature-value pairs, we give the definition of Object. Finally, we introduce other basic concepts for the development of the framework of Object-oriented Recommender System (ORS).

Definition 2.1. A Travel Package is a fixed suite of integrated travel information provided by a travel company for the tourists, such as some landscapes, the travel days and the price.

We explore a real-world travel data set provided by a travel company in China. From this data set, we extracted 23,351 useful records from 5,211 tourists for 908 travel packages from the year of 2000 to 2010, and each tourist has traveled at least two different packages. Note that all the following discussions are based on the statistical analysis of this real-world data set. There are some unique characteristics of the travel



(a) The average cost of tourists with different age and gender.

(b) The travel season of tourists with different age.

Fig. 1. The relationships among different features.

Table I. Number of travel packages and new packages each year.

Year	2005	2006	2007	2008	2009	2010
Packages	21	98	286	298	269	176
New Packages	21	82	243	229	207	126

data, some of which have been briefly illustrated in [Liu et al. 2011; Liu et al.]. First, it is very sparse. On average, each tourist has traveled only 4 times and only 0.49% of the entries in the corresponding tourist package matrix are non-zero. The extreme sparseness of the data raises the challenges for traditional recommendation methods, such as the collaborative filtering which needs to discover enough and trustable similar users or items. It is also one of the reasons that we exploit the additional context information for improving travel package recommendation.

Next, the choice of the tourists is highly dependent on the attribute correlations between tourists and travel packages. For example, tourists with different age and gender also have different affordable prices. In Fig. 1(a), we can know that male tourists cost more money than female tourists on average, and tourists with different ages usually have different spending patterns. Thus, both the gender and the age of the tourists affect their choice when the tourists go to travel. As Fig. 1(b) shows, young tourists aged from 15 to 24 have half of travel records occurred in summer maybe because of the spare time in summer holiday. Also according to Fig. 1(b), the percentage of elders traveling in Fall is increasing with age, perhaps because Fall has more comfortable weather than other seasons. Let's consider a simple example, if there is a 20-years-old girl named Alice, based on Fig. 1(a) and Fig. 1(b), she should have higher possibility to choose a cheap short trip in summer than a luxury travel in winter. Similarly, it is easy to find that there are many other features affect the choices of tourists and the recommendation effects. Therefore, it is important to systematically incorporate these features and effectively use them for the travel package recommendation.

Finally, the travel data have much stronger time dependence. Indeed, Table I shows that most of the travel packages are new, where a new package means a package which does not previously exist and is recently added into the system. As illustrated in Table I, tourists like to choose novel packages, so that each year the travel companies create new travel packages to replace the old ones. Without traveling records, the only way to recommend these new packages is to exploit their content/context information.

In summary, for a specific tourist, the travel package chosen by him is dependent on both the attributes of himself, e.g., age and gender, and the features of that travel

Table II. A toy example about tourists traveling in Hongkong.

	Age:Young	Age:Middle
Price:Cheap	The common choice of young tourists.	Few of middle-aged choose this.
Price:High	Few of young tourists choose this.	The common choice of middle-aged.

package, e.g., price and travel season. A toy example is shown in Table II, assuming that a group of tourists have the interest traveling to Hongkong. Without the context information (e.g., package features and tourist attributes), it is hard to find out the reason they go to Hongkong. By considering the feature “Age” and “Price”, according to Fig. 1(a), we can infer that young tourists are interested in the cheap trip in Hongkong while middle-aged may enjoy more luxury tours. Thus, we can provide appropriate services through recommending different packages to tourists with different attributes. We conclude that the context information provided by feature-value pairs could describe the interests of the tourists more precisely. However, analyzing each feature case-by-case is not an optimal choice as there are so many different types of features. Generally, in this paper, we define **Feature-value Pair** as the unified expression for the feature and its corresponding value of both tourists and travel packages.

Definition 2.2. A Feature-value Pair is a unified expression of the attribute/feature and its corresponding value of an instance in recommender systems.

In this way, a tourist or a package is just an encapsulation of some feature-value pairs. We further assume that the interactions between tourists and packages are decided by their feature-value pairs. Therefore, the instances in recommender system, i.e., tourist, package and travel record can be abstracted to each Object by a collection of feature-value pairs. Formally, we define **Object** in recommender systems as follows:

Definition 2.3. An Object in recommender systems is a collection of feature-value pairs, which is an abstract description of an instance in the real world.

Generally, Object in recommender systems can be user, item (i.e., package in this paper) or relationship between users and items. A travel record shows that a user chose a package at a certain time, so that the travel record’s feature-value pairs are built from the feature-value pairs of tourist, package, and the travel time. For example, Alice, who is referred in Section 1, one of her travel record can be represented as {Name:Alice, Age:20, Gender:female, Days:1, Area:Hong Kong, Season:summer, Year:2011}, where each feature-value pair is shown in the “feature:value” style.

Based on the above definitions, all types of features can be represented simultaneously and uniformly, and new feature-value pairs could be added dynamically and naturally. Meanwhile, in this way, the contributions of all the feature-value pairs for each Object can be computed, rather than considering them case-by-case. It is obvious that the way to discover the relationships/similarities among Objects is the most important step. A common method is to directly compute the similarity of their feature-value pairs, but it ignores the possible relations among these feature-value pairs. For this reason, we propose two different models, the Object-oriented Topic Model (OTM) and the Object-oriented Bayesian Network (OBN), to capture both the relationships among Objects and the latent relations among these feature-value pairs. Both of the models are proven to be effective in the experiments. Note that these two models also have their own unique characters, OTM can help developers select useful features, while OBN has better recommendation results and consumes fewer computing resources. In

Table III. Mathematical notations.

Notation	Description
$U = \{U_1, U_2, ..., U_i, ..., U_{\|U\|}\}$	the set of tourists
$P = \{P_1, P_2, ..., P_{\|P\|}\}$	the set of packages
$Y = \{Y_1, Y_2, ..., Y_j, ..., Y_{\|Y\|}\}$	the set of years
$T = \{T_1, T_2, ..., T_k, ..., T_{\|T\|}\}$	the set of topics
$F = \{F_1, F_2, ..., F_m, ..., F_{\|F\|}\}$	the set of features
$V = \{V_1, V_2, ..., V_l, ..., V_{\|V\|}\}$	the set of values

the following three sections, we first propose the OTM and OBN models respectively, and then present the whole working process of the ORS framework based on OTM or OBN. For the purpose of illustration, Table III lists some mathematical notations.

3. OBJECT-ORIENTED TOPIC MODEL

In this section, we introduce the way to represent the Objects by a topic model for identifying correlations and relationships among feature-value pairs. There are several reasons that we propose a topic model. First, topic model can effectively explore tourists' interests from the historical travel records [Liu et al. 2011], i.e., it helps understand the Objects by their latent topics. Meanwhile, following the strategies in [Liu et al. 2011; Blei et al. 2003; Rosen-Zvi et al. 2004; McCallum et al. 2007; Bao et al. 2010], the similarity between different Objects (e.g., packages and tourists) can be measured.

In recommender systems, the recommendation list is dependent on the interests of the given user. Because user's interests are usually implicit, researchers can only explore them from the historical records [Adomavicius and Tuzhilin 2005]. Specifically, for a given tourist, his travel interests will be explored from the prior travel records. As discussed in Section 2, the travel record can be also encapsulated into an Object, which is a collection of feature-value pairs. Meanwhile, a tourist may have traveled once or many more times, so his records include a number of different feature-value pairs. Note that each tourist is also an Object in the ORS framework, therefore the feature-value pairs representing the tourist are composed by three parts: the personal profiles of the tourist, the attributes of the travel packages traveled by the tourist, and other feature-value pairs recorded in his travel history. Then, the problem becomes how to measure the travel interests of the tourists by these feature-value pairs. Considering that an Object is a collection of feature-value pairs, and a document in the topic model is a collection of words (i.e., bag-of-words) [Blei et al. 2003], thus the idea of projecting words into latent topics by topic models for finding the correlations between words can be also adopted for representing Objects and discovering feature-value pair correlations. Then, the tourists' travel interests can be mined, and the similarity between tourists will be computed.

Actually, topic models are generative models that have been successfully used for document modeling [Blei et al. 2003; Rosen-Zvi et al. 2004; McCallum et al. 2007]. In addition, Bao et al. [Bao et al. 2010] developed a LDAC (Latent Dirichlet Allocation on Context) model for mobile user modeling where they chose the similar feature-value pair representations. Recently, [Agarwal and Chen 2010; Liu et al. 2011; Liu et al. 2012] indicate that topic models can be also used for recommender systems. Generally, topic models assume that there are several topics for a corpus D, and a document d in D can be viewed as a bag of words $w_{d,i}$ which are generated by these topics. Intuitively, if we take the feature-value pairs as words, the Object(e.g., tourist) as the bags of

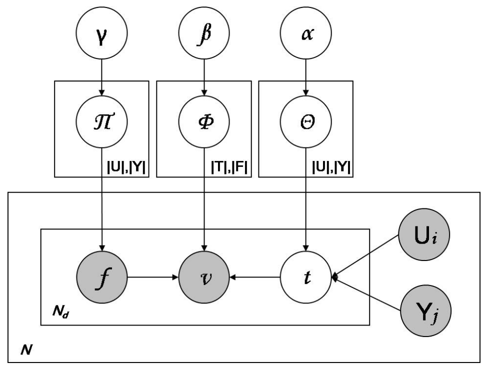


Fig. 2. A graphical representation of OTM.

feature-value pairs, and the latent travel interests as topics, we can take advantage of topic models to learn tourists' implicit interests.

However, there are two key differences between traditional topic models and our Object modeling. First, the words in traditional models are just dependent on the distribution of the topics. In our scenario, the occurrences of the values are dependent on both the latent topics and the corresponding features. Actually, for generating a feature-value pair, both the topic and the feature will be decided first, and then the corresponding value of the feature can be generated by the joint distribution of the topic and the feature. For instance, assuming that there are 100 tourists, 40 of them have traveled in Hongkong and 50 of them are young tourists, and 30 of young tourists have traveled in Hongkong. Without the feature "Age", it is induced that tourists have a probability of 0.4 on the topic "go to Hongkong". If we take the feature "Age" into consideration, the probability of young tourists enjoying "go to Hongkong" increases to 0.6. Therefore, for a given tourist, if the related feature (e.g., age) and the corresponding value (e.g., young) are known, the estimation of his latent interest can be more accurate. Second, because the interests of the tourists are time-sensitive, we consider the annual change of tourists' preferences. For example, if we make recommendations for a tourist in the year of 2010, it is inappropriate to recommend the package that he may like in 2007. Thus, we split the travel records by year for understanding and emphasizing the annual travel preferences of the tourists. Along this line, we extend the existing topic models [Bao et al. 2010; Liu et al. 2011] for the Object modeling.

Based on the above discussion, we develop the Object-oriented Topic Model (OTM), where feature-value pairs are treated uniformly (except for the feature "year", and the reason has been given previously). Mathematically, the generative process corresponding to the hierarchical Bayesian model of OTM is shown in Fig. 2, where shaded and unshaded variables indicate observed and latent variables respectively.

In OTM, a specific document d_{ij}, one of the N documents in the travel record set D, is decided by U_i and Y_j, and it contains all the travel information (represented by a collection of N_d feature-value pairs) that tourist U_i traveled in year Y_j. As a result, the topic distribution of document d_{ij} represents the interests of tourist U_i in year Y_j. For finding the latent topics in the corpus D, we first consider the document generation process. Specifically, we take the generation of the n-th feature-value pair (f_n, v_l) for d_{ij} as an example. This process is as follows:

(1) Choose $\theta_{ij} \sim Dirichlet(\alpha)$

(2) Choose $\phi_{k,f_n} \sim Dirichlet(\beta)$

(3) Choose $\pi_{ij} \sim Dirichlet(\gamma)$

(4) For the n-th feature-value pair (f_n, v_l) in d_{ij}:
   (a) Topic t_k is generated from θ_{ij};
   (b) Feature f_n is generated from π_{ij};
   (c) The value v_l of f_n is generated from the distribution ϕ_{k,f_n}

Similar to LDA model [Blei et al. 2003], given the parameters α, β and γ, we can obtain the marginal distribution of a document d_{ij} with N_d feature-value pairs:

$$P(d_{ij}|\alpha,\beta,\gamma) = P(U_i)P(Y_j)\int\int\int P(\theta_{ij}|\alpha)P(\pi_{ij}|\gamma)\prod_{k=1}^{|T|}\prod_{m=1}^{|F|}P(\phi_{km}|\beta) \prod_{n=1}^{N_d}\sum_{k=1}^{|T|}\left(P(t_k|\theta_{ij})P(f_n|\pi_{ij})P(v_n|\phi_{k,f_n})\right)d\pi_{ij}d\phi d\theta_{ij}$$

where $P(U_i)$ and $P(Y_j)$ stand for the probability of choosing tourist U_i and year Y_j, respectively. As these two values are constants and they can be directly computed from the travel records, in the following we omit them for better illustration. Then, taking the product of the marginal probabilities of single documents, we can obtain the probability of the entire travel record set D:

$$P(D|\alpha,\beta,\gamma) = \int\int\int\prod_{i=1}^{|U|}\prod_{j=1}^{|Y|}P(\theta_{ij}|\alpha)\prod_{k=1}^{|T|}\prod_{m=1}^{|F|}P(\phi_{km}|\beta)\prod_{i=1}^{|U|}\prod_{j=1}^{|Y|}P(\pi_{ij}|\gamma) \prod_{i=1}^{|U|}\prod_{j=1}^{|Y|}\prod_{n=1}^{N_d}\sum_{k=1}^{|T|}\left(P(t_k|\theta_{ij})P(f_n|\pi_{ij})P(v_n|\phi_{k,f_n})\right)d\pi d\phi d\theta$$

For the inference purpose, we exploit the Gibbs sampling method [Griffiths and Steyvers 2004], a form of Markov chain Monte Carlo, to extract a set of topics from a large set of traveling records. During Gibbs sampling, the generation of each feature-value pair token for a given travel record depends on the topic distribution of the corresponding tourist-year pair and the value distribution of the topic-feature pair. Finally, the estimations of θ, π and ϕ given the training set can be calculated by:

$$\theta_{ijk} = \frac{\alpha_k + n_{ijk}}{\Sigma_{t=1}^{T}(\alpha_t + n_{ijt})}$$

$$\pi_{ijm} = \frac{\gamma_m + n_{ijm}}{\Sigma_{f=1}^{F}(\gamma_f + n_{ijf})}$$

$$\phi_{kml} = \frac{\beta_m + n_{kml}}{\Sigma_{v\in f_m}(\beta_v + n_{kmv})}$$

where n_{ijk} is the number of the feature-value pair tokens assigned to topic T_k and tourist-year pair (U_i, Y_j), n_{ijm} is the number of the corresponding feature of the m-th feature-value pair in document d_{ij} decided by U_i and Y_j, and n_{kml} is the number of the value v_l assigned to topic T_k and feature f_m.

During Gibbs sampling, the more frequently two feature-value pairs co-occurred, the more likely for them to be assigned by the same topic. Then, after the Gibbs sampling, all the tourists can be represented as different topic distribution vectors. By computing the similarity of their topic distribution vectors, we can find the similarities among the tourists. However, we should note that, the inference of the OTM model is very time-consuming, and the computation cost will be higher if the travel records become larger. Since the travel topics evolve very slowly, we can update the inference process periodically in an off-line manner.

In addition, there are other benefits of the OTM model. First, we can find the important feature-value pairs for each topic. One step further, we make a detailed analysis in Section 7 about how to find the most important features by OTM model. Also, it should be pointed out that new features can be added without any extra burden. Since the feature-value pairs are processed as the words and the number of words nearly have no effect on topic models, this means OTM model can harmony and almost unlimitedly import additional information. At last, similar to traditional topic models, the topics extracted by OTM are composed by feature-value pairs, and thus these topics can be visualized, explainable and easy to understanding.

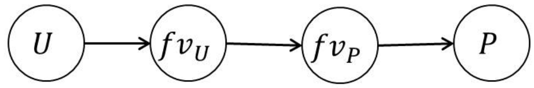
4. OBJECT-ORIENTED BAYESIAN NETWORK

The OTM model shows its capability in extracting hidden tourist interests as topics from additional information. However, there are still limitations to apply OTM in practical applications. The first and most important one is the high time consumption of estimating a topic model. Although the training process can work off-line, it still needs much computations as tourist number increases. These limitations motivate us find simpler and more efficient method for discovering the relationship among feature-value pairs. In this section, we propose an Object-oriented Bayesian Network (OBN) model. Similar to OTM, the OBN model also can be used for finding the relationship among tourists and packages. However, OBN model does not explain the reason that tourists choosing packages by extracting some latent topics. Alternatively, it builds a Bayesian Network[Breese et al. 1998] for tourists, packages and feature-value pairs to directly infer the probability of tourists' co-travel, i.e., OBN builds a hybrid Bayesian network, where the nodes can be the tourists, packages or feature-value pairs.

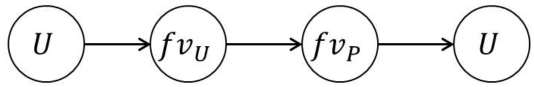
Bayesian models have been used for recommendation before. For instance, [Breese et al. 1998] represented each item as a node in a Bayesian network, where the states of each node correspond to the possible rating values for that item. Similarly, [Harvey et al. 2011] also proposed a Bayesian latent variable model for rating prediction. To the best of our knowledge, both of these Bayesian models are used for rating estimations rather than ranking prediction or travel package recommendation.

Before introducing the OBN model, we consider some simple scenarios. For example, as we have discussed in Section 2, we can draw assumptions like "40% of elders travel in Fall" and "Alice travel to Hongkong with a probability of 90%" by analyzing statistical data from the travel logs of elders or Alice. It is also easy to get the similar results for other tourists from the travel logs. Actually, for a specified tourist, the travel package chosen by him is dependent on both his personal attributes, e.g., age and gender, and the features of the travel package, e.g., price and season. Thus, we can consider the choices between tourists and packages as the choices of feature-value pairs. Along this line, given the travel records, we learn the chosen probability between tourists and packages by maximum likelihood estimation (MLE). Here, we use $P(fv_p|fv_u)$ denote the probability of tourists with feature-value pair fv_u choosing packages with fv_p.

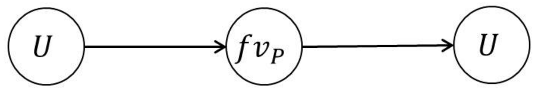
$$P(fv_p|fv_u) = \frac{\text{co-occurrences time of } fv_p \text{ and } fv_u}{\text{occurrences time of } fv_u}$$



(a) The direct inference from tourists to travel packages.



(b) The global probability of the co-travel inference.



(c) The personal probability of the co-travel inference.

Fig. 3. The Object-oriented Bayesian Network.

Treating tourists as one class of Object, travel packages as another, these two classes of Objects could link to each other through their features. Thus, a two-level Bayesian network as shown in Fig. 3(a) can be built in which feature-value pairs fv_u belonging to tourists are the nodes of the first level, and fv_p are the second. If we want to know how likely is tourist U_i to choose package P_j, we can compute the probability $P(P_j|U_i)$ by the following equation:

$$P(P_j|U_i) = \sum_{fv_u \in U_i} \sum_{fv_p \in P_j} P(fv_p|fv_u)$$

However, there are still some limitations. First, $P(P_j|U_i)$ share the same value for different tourists as long as they have the same feature-value pairs. Second and more importantly, directly calculating $P(P_j|U_i)$ ignores the influence from other tourists, which is the key factor that has been addressed by the idea of collaborative filtering[Resnick et al. 1994; Sarwar et al. 2001].

Thus, we propose the Object-oriented Bayesian Network (OBN) model to directly infer the co-travel probabilities among tourists, rather than the chosen probability from tourists to travel packages. For solving the first limitation, we can define $P(fv_p|U_i)$ denoting the probability of tourist U_i choosing packages with feature-value pair fv_p.

$$P(fv_p|U_i) = \frac{\text{the time of } U_i \text{ choose a package with } fv_p}{\text{travel time of } U_i}$$

$P(fv_p|U_i)$ is a personal probability for each tourist. In the OBN model, tourists rather than feature-value pairs, are the first-level nodes.

For the second limitation, instead of the direct inference from tourists to travel packages, we consider to calculate the co-travel probability $P(U_i|U_j)$ i.e., the probability that when U_j traveling, U_i also travels with him/her. We can calculate a weighted sum as final probability which contains personal and global influence for each tourist.

$$P(U_i|U_j) = \lambda \sum_{fv_u \in U_i} \sum_{fv_p} P(fv_u|fv_p)P(fv_p|U_j) + (1-\lambda) \sum_{fv_p} P(U_i|fv_p)P(fv_p|U_j)$$

where the conditional probability $P(U_i|fv_p)$ means the probability that tourist U_i appears in the travel records of the packages having feature-value pair fv_p. In the right of above equation, the first part is the global probability that U_j travels with tourists having same feature-value pairs with U_i, the second part is the personal probability that U_j travels with U_i. and λ is the weight, i.e., $\lambda \in [0, 1]$.

For collaborative filtering, $P(U_i|U_j)$ can be considered as the similarity of U_i and U_j, $Sim(U_i, U_j) = P(U_j|U_i)$. It should be noted that $Sim(U_i, U_j) \neq Sim(U_j, U_i)$. Fig. 3(b) and Fig. 3(c) show the global and personal probability of the co-travel respectively.

It is easy to understand the OBN model which is a simple as well as intuitive Bayesian network, and we can infer the relationship among feature-value pairs or tourists just using the co-occurrence time of them. It can be implemented easily, and updated in real time by just modifying the co-occurrences time. Therefore, this is a possible model to be adopted by large-scale practical recommender systems.

5. THE ORS FRAMEWORK

Having said that in this paper we propose a recommendation framework, named *Object-oriented Recommender System* (ORS), which is very flexible and effective in terms of incorporating multiple types of additional context information represented by feature-value pairs. In this section, we show the way to apply the ORS framework for travel package recommendation, so as to take full advantages of the feature-value pairs and the Object. We hope ORS could help developers attract the tourists before they make a travel decision, e.g., by email marketing. Generally speaking, the working process of the ORS framework is as follows:

(1) Extracting feature-value pairs from the raw travel records, and segmenting continuous values to category values for building Objects.
(2) Encapsulating instances (e.g., tourists) to be Objects, with the feature-value pairs extracted from the travel records and profiles;
(3) Developing models, i.e., OTM or OBN, for discovering similarities among tourists;
(4) Generating the recommendation/ranking results by the annual collaborative filtering method according to the similarities discovered/output by OTM or OBN;
(5) Refining the recommendation list, i.e., adding the new packages into the list by computing similarities with the candidate packages generated previously.

Fig. 4 shows the flowchart of ORS framework, where the OTM model and OBN model are interchangeable because they both output the tourist similarities. Since the details of OTM model and OBN model have been described in Section 3 and Section 4, in the following, we introduce the techniques used in other steps.

5.1. Feature Selection and Segmentation

To describe the Object, seven major features are extracted from the raw travel data, these features are **Age** and **Gender** from tourists, **Area**, **Price** and **Days** from travel packages, and **Season** and **Year** from travel records. Each feature represents one contextual characteristic of an Object. The age and gender describe the personal attributes, area is about geographical location, price and days are about the financial and time cost respectively, while season and year show clearly the travel time, and season also suggests the macro-climate conditions. Then we consider the value range of these features in Table IV. While the sematic values of gender, days and year are easy-to-understand, in the following, we show the technical way to segment the continuous values of the rest features into categorical values.

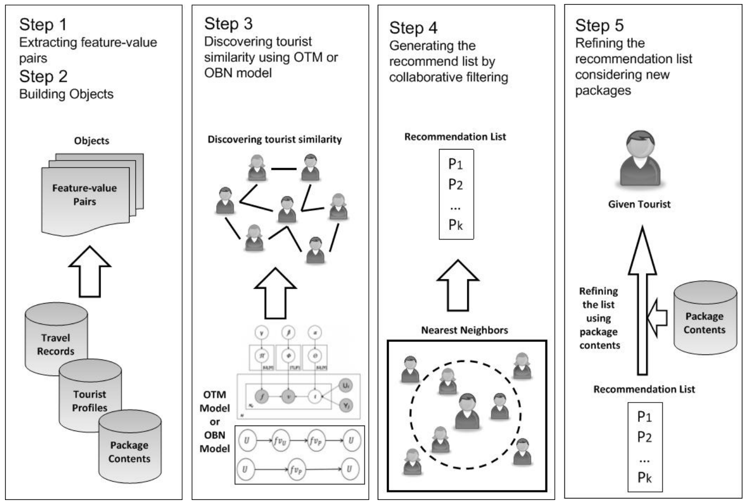


Fig. 4. The ORS framework using OTM or OBN model.

Table IV. Features and the range of values.

Feature	Values
Age	Child, Young, Middle, Old
Gender	Female, Male
Area	SC, CC, NC, EA, SA, OC, NA
Price	Very low, Low, Medium, High, Very high
Days	1, 2, 3, ..., 12
Season	Spring, Summer, Fall, Winter
Year	2004, 2005, ..., 2010

Area. We can not simply view each city or province as an area because it will be too detailed and lead to the over-fitting problem. In contrast, a coarse partition of the space will lead to the loss of spatial information. Thus, we divide the entire location space in the data set into 7 big areas according to the travel area segmentations provided by the travel company, which are South China (SC), Center China (CC), North China (NC), East Asia (EA), Southeast Asia (SA), Oceania (OC) and North America (NA), respectively. The area segmentation results are shown in Table V.

Season. We assume that the travel packages have a relatively stable distribution in each season. Then, we use an information gain based method [Fayyad and Irani 1993] to get the season segmentation. The information entropy of season S^P is given by $Ent(S^P) = -\sum_{i=1}^{|S^P|} p_i log(p_i)$, where $|S^P|$ is the number of different packages in S^P and p_i is the proportion of package P_i in this season. Initially, the entire year is viewed as a big season and partitioned into several seasons in a recursive binary way. In each iteration, we use the weighted average entropy (WAE) to find the best split:

$$WAE(i; S^P) = \frac{|S_1^P(i)|}{|S^P|} Ent(S_1^P(i)) + \frac{|S_2^P(i)|}{|S^P|} Ent(S_2^P(i))$$

where $S_1^P(i)$ and $S_2^P(i)$ are two sub-seasons of season S^P when being split at the i-th month. The best split month induces a maximum information gain given by $\triangle E(i)$,

Table V. Area segmentation result.

Area	Provinces/Countries
SC	Guangdong, Guangxi, Taiwan, Yunnan, Hong Kong, Fujian, Hainan, Macau
CC	Jiangxi, Guizhou, Sichuan, Hunan, Zhejiang, Jiangsu, Shanghai, Chongqing, Hubei, Anhui
NC	Shanxi, Henan, Hebei, Heilongjiang, Jilin, Liaoning, Gansu, Neimenggu, Ningxia, Xizang, Qinghai, Beijing, Tianjing, Shaanxi, Xinjiang, Shandong
EA	Japan, South Korea
SA	Singapore, Malaysia, Thailand, Brunei
OC	Australia, New Zealand
NA	USA

which is equal to $Ent(S^P) - WAE(i; S^P)$. As a result, January and February belong to Winter, March to May are Spring, June to September are Summer, and the rest months are Fall. The result is consistent with the priori knowledge that all the tourists in this data are from the Southern China.

Price. Similar to season segmentation, we divide the prices of the packages based on the variance of prices in the travel data [Yuan et al. 2010]. The split result is as follows, very low price is $(0, 243]$, low is $(243, 664]$, medium is $(664, 1, 740]$, high is $(1, 740, 5, 478]$, and very high price is higher than 5,478, the unit is RMB (CNY). The adopted methods for area, season and price segmentations are similar to that in [Liu et al. 2011].

Age. We divide the ages of the tourists using the similar method as the price segmentation, and the age segmentation result is as follows. Child: < 16, Young: $>= 16$ and < 30, Middle: $>= 30$ and < 60, Old: $>= 60$.

5.2. Generating the Initial Recommendation List

Based on extracted feature-value pairs of each Object, we use the OTM or OBN model to obtain the relationships among tourists. In this paper, the whole travel records of a tourist in one year is treated as an Object. Thus, we can compute the similarity between each tourist in the specific year, and collaborating filtering can be adopted for generating the personalized candidate package set for each tourist.

Intuitively, in the collaborative filtering, for a given user, we recommend the items that are preferred by the users who have similar interests with him. However, the travel package recommendation is more complex than the traditional ones. For a given tourist, we should find his nearest neighbors by ranking their similarity values, and recommend the packages that are liked by the neighbors in the specific year. Thus, the packages, which are favored by these neighbors but have not been traveled by the given tourist, can be selected as candidate packages which form an initial recommendation list, and they are ranked by the probabilities computed by the collaborative filtering.

For the OTM model, we have obtained the annual topic distribution for each tourist and they are represented in vectors with the same length. For computing the similarity between tourist U_m and U_n in year Y_j, we use *Correlation Coefficient* [Resnick et al. 1994], a simple but effective technique:

$$Sim_{Y_j}(U_m, U_n) = \frac{\sum_{k=1}^{K}(\theta_{mjk} - \bar{\theta}_{mj})(\theta_{njk} - \bar{\theta}_{nj})}{\sqrt{\sum_{k=1}^{K}(\theta_{mjk} - \bar{\theta}_{mj})^2}\sqrt{\sum_{k=1}^{K}(\theta_{njk} - \bar{\theta}_{nj})^2}}$$

where $\bar{\theta}_{mj}$ is the average topic probability for the tourist-year pair (U_m, S_j). If the given tourist U_m has never traveled in Year Y_j, then his/her total topic distribution ϑ_m^U is used as an alternative throughout this paper.

For the OBN model, we have $Sim(U_m, U_n) = P(U_m|U_n)$ from Section 4. We consider **Age**, **Gender** and **Year** as the features belonging to tourists (i.e., fv_u), **Area**, **Price**, **Days** and **Season** to packages (i.e., fv_p). Thus, the annual similarity for tourists can be calculated, when the year is given

$$Sim_{Y_j}(U_m, U_n) = P_{Y_j}(U_m|U_n) = \lambda \sum_{fv_u \in U_m} \sum_{fv_p} P(fv_u|fv_p)P(fv_p|U_n) + (1-\lambda)\sum_{fv_p} P(U_m|fv_p)P(fv_p|U_n)$$

When calculating $P_{Y_j}(U_m|U_n)$, we should just consider two tourists' co-travel time in the given year Y_j.

5.3. Refining the Recommendation List

We introduce the way to refine the recommendation list so as to recommend new packages for alleviating the cold-start problem. For the travel data, as we have explored in Section 2, new packages are created every year and most of the active packages are the new ones. Since the packages are composed by the landscapes, and most of the landscapes will keep in use even after the original package has been discarded [Liu et al. 2011], we can compute the similarity between any pair of packages as follows

$$Sim(P_i, P_j) = \frac{|L_{P_i} \cap L_{P_j}|}{|L_{P_i} \cup L_{P_j}|}$$

where L_{P_i} means the set of landscapes composing the package P_i. We propose to compute the similarity between the new package and the given number (e.g. 10) of candidate packages in the top of the recommendation list. Then, new packages are added into the recommendation list and the ranks of these new packages are based on the average probabilities of the similar candidate packages. Finally, after removing the packages which are no longer active, we will have the final recommendation list.

We can see that the ORS framework follows the hybrid recommendation strategy and combines many factors together. Thus, the challenges mentioned in Section 2 could be addressed, for instance, the data sparsity is alleviated by importing additional context information while the recommendation effects of these context information is learnt systematically and cost-effectively (by either OTM or OBN), and the time dependence is considered by including new travel packages into the recommendation list. In this way, the ORS framework is an open and effective framework in terms of incorporating additional context information as feature-value pairs.

6. EXPERIMENTAL EVALUATION

In this section, we evaluate the performances of the ORS framework. For convenience, we use ORS-OTM and ORS-OBN stand for the ORS framework with the OTM and OBN model separately. Because the OTM model is a topic model, we also demonstrate the predictive power of the OTM model measured by the perplexity value, and the understanding of the topics extracted by the OTM model.

6.1. The Experimental Setup

Experimental Data. The data set was divided into a training set and a test set. The last travel record of each tourist was chosen to be part of the test set, and the remaining records were used for training. In total, there are 5,211 tourists, 18,140 travel records for 805 packages in the training set, and 5,211 travel records and 601 travel packages for testing. There are 103 new packages traveled by 387 tourists in test set.

Benchmark Methods. For the recommendation evaluation, we compare with the following methods:

— Three similar frameworks based on topic models: LDA-P, LDA-L,LDA-F, which take the packages, landscapes, and feature-value pairs as words respectively in LDA model [Blei et al. 2003]. After the LDA model has been trained, the user based collaborative filtering method is used for recommendation and the user similarities are based on the vector similarity of the latent topic distributions.
— Meanwhile, we implemented the user based collaborative filtering method (UCF-P) [Resnick et al. 1994], the item based collaborative filtering method (ICF-P) [Sarwar et al. 2001] and the hybrid collaborative filtering method (HybridCF) [Li et al. 2005] for collaborative filtering.
— Since UCF-P and ICF-P only consider package level information, for making a more fair comparison, we implemented two similar methods based on feature-value pairs (UCF-F, ICF-F). Specifically, in UCF-F or ICF-F, we just calculate the set (collection of feature-value pairs) similarity between tourists or packages for collaborative filtering. And in UCF-P or ICF-P, the set is a collection of packages or tourists.
— We also compare ORS with the Cocktail recommendation approach based on TAST model [Liu et al. 2011].
— At last, we implemented the Always-Choose-Most-Popular method (MostPop), and a Bayesian Network classifier (UIBayes) based on Fig. 3(a). The UIBayes model uses tourist and package features for input and the output is the probability that given tourist chooses this package.

All the above methods (UCF-P, ICF-P, UCF-F, ICF-F, LDA-P, LDA-L, LDA-F, HybridCF, Cocktail, MostPop, UIBayes) are the benchmarks.

6.2. Recommendation Evaluation Metrics

We adopt Degree of Agreement (DOA), Top-K, Normalized Discounted Cumulative Gain (NDCG) and Novelty as the evaluation metrics. All of them are commonly used, and they characterize the recommendation results from different perspectives. Noting that all the metrics are the bigger the better. Also, we conduct a user study and let volunteers give rate to the recommendations.

DOA measures how much item pairs ranked in the correct order [Fouss et al. 2007; Liu et al. 2012]. Let N_{U_i} denote the set of items that neither in the training set nor in the test set of tourist U_i, and T_{U_i} means the set of items that in the test set. The function $correctOrder_{U_i}(P_j, P_k)$ is 1 if the predicted rank of P_j is higher than P_k, otherwise 0. Then, the individual DOA for user U_i can be defined as follows:

$$DOA_{U_i} = \frac{\sum_{P_j \in E_{U_i}, P_k \in N_{U_i}} correctOrder_{U_i}(P_j, P_k)}{|E_{U_i}| \times |N_{U_i}|}$$

An ideal ranking list will lead to a 100% DOA, and we use the average DOA of all DOA_{U_i} as the final metric.

Top-K indicates the effectiveness of the recommendation from a cumulative way [Koren 2008]. Let H_i denotes the number of hits to the test set of user U_i, K means the selected top $K\%$ travel packages of all, the Top-K are defined as follows:

$$Top-K = \frac{1}{|U|} \sum_{U_i \in U} \frac{H_i}{|P| \times K\%}$$

NDCG evaluates the quality of a ranking result in information retrieval by assigning graded content relevance judgments [Xie et al. 2010; Liu et al. 2011]. The NDCG metric assumes that packages with higher correlation should have higher ranks in the recommendation list. In this paper, we compute the content relevance of two travel packages $R(P_i, P_j)$ as $\frac{|L_{P_i} \cap L_{P_j}|}{|L_{P_i}|}$, where L_{P_i} means the set of landscapes composing

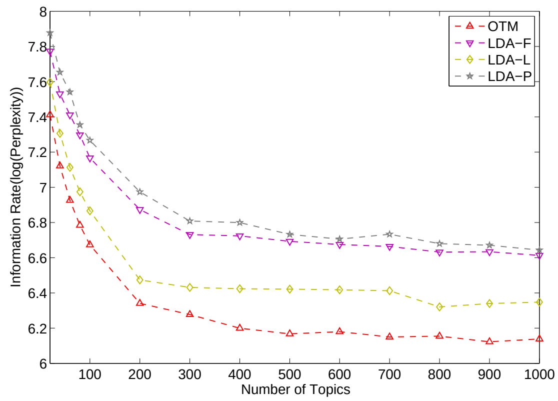


Fig. 5. Perplexity comparison.

the package P_i. Thus, the NDCG value at k-th position of the ranking list for a given tourist can be computed by:

$$NDCG@k = \frac{RL@k}{IRL@k}, \quad RL@k = R(P_t, P_1) + \sum_{i=2}^{k} \frac{R(P_t, P_i)}{log_2(i)}$$

where P_t is the test package, RL is the ranking list and IRL is the ideal list.

Novelty can be seen as the ability of a recommender to introduce users to items that they have not previously experienced before in real life [Zhou et al. 2010]. We measure novelty with a metric introduced in [Zhou et al. 2010]:

$$Novelty@k = \frac{1}{|U|} \sum_{u \in U} \sum_{i \in R_{uk}} \frac{\log_2(|U|/c_i)}{k}$$

where $\log_2(|U|/c_i)$ gives self-information of item i, c_i is the number of the choices of item i in training set. For new packages, we set $c_i = 1$. R_{uk} gives the top k recommended items for user u. In the experiments, k us ranged from 1 to 30.

User study. Since high ranking accuracy may still lead to the low-quality recommendation, we also conducted a user study and collected some extra data (volunteer feedbacks) to make the evaluation more reliable.

6.3. Perplexity Comparison for OTM

The topic models are often evaluated by perplexity for measuring the goodness of fit. The lower perplexity a model is, the better it predicts the new documents [McCallum et al. 2007]. When the tourist U_i and the travel year Y_j are given, the perplexity of a unseen travel record d_{ij} including feature-value pairs $P_{d_{ij}}$ can be defined as follows:

$$Perplexity(P_{d_{ij}}) = exp(-\frac{logP(P_{d_{ij}}|U_i, Y_j)}{|P_{d_{ij}}|})$$

For the fitness purpose, we compare the OTM model with three topic models LDA-P, LDA-L and LDA-F. We choose the fixed Dirichlet distributions with $\alpha = 50/T$ and $\beta = 0.1$ for these topic models, and these settings are also used in the existing works [Griffiths and Steyvers 2004; McCallum et al. 2007; Liu et al. 2011]. In the experiments, the Markov chains were run with different initializations, and the samples at

Table VI. Topics illustration with different feature-value characteristics.

Year	2008			2009		
Topic	20	35	44	11	12	55
Age	Young	Middle	Young	Middle	Middle	Old
Gender	Male	Female	Female	Female	Female	Male
Area	NC	SA,OC	SC	NC	SC	SA
Price	Medium	Very high,high	Very low	High	Very low	High
Days	8	10,12	1	5	1	5
Season	Fall	Summer	Summer	Fall	Fall	Spring

the 1001th iteration were used to estimate θ and ϕ. The average information rate (logarithm of perplexity) with different numbers of topics on the data set is shown in Fig. 5. As shown in the figure, OTM has significantly better predictive power than three other models. Among them, LDA-P performs the worst, that is because the information that LDA-P takes into consideration is the least. In contrast, LDA-L performs much better than LDA-P and this again demonstrates the fact that landscapes are more useful and important than the packages themselves [Liu et al. 2011].

6.4. Topics Identified by the OTM Model

Here, we mainly focus on studying the relations between the topics and their characteristics from the Objects, for better understanding the mined travel topics.

Table VI shows the feature-value pairs with the highest probability from six topics in the OTM model trained with 200 topics. We choose these six topics from two years period, so that we can see the topics are changed over time but still keep some similarities. For example, let us look at Topic 20 in 2008 and Topic 11 in 2009, they are all about the area of North China, but the price is higher and the days is shorter in 2009. Also, if we focus on a certain tourist group, such as middle-aged female in Topic 11 and 12, we can find that this group have two different types of consumption tendency. Then, different groups with different characteristics also have different travel interests. As shown in Topic 35 and 55, some older men tend to travel in Southeast Asia in Spring, and some middle-aged women like to cost more time in Summer also in Southeast Asia. The above observation agrees with the statistical results as shown in Fig. 1(b). Based on the correlations among the feature-value pairs, all the topics can be understood as the latent interests of tourists. This suggests that the OTM model can precisely capture the user preferences.

6.5. The Recommendation Performances

In this subsection, we present the performance comparison on recommendation effects between ORS and the benchmark methods. For the purpose of comparison, we fix topic=200 for LDA-F, LDA-L, LDA-P and ORS, because the variances of perplexity become less obvious since then, as shown in Fig. 5. We also set the nearest neighbor size of UCF as 1000, and 500 for ICF. For the ORS-OBN method, we set the weight λ as 0.5, i.e., the global and personal probability are of equal importance. This compromise value is based on the results shown in Fig. 6. As shown in Fig. 6, when the weight λ changes from 0 to 1, DOA remains approximately constant, while Topk-10 and $NDCG@10$ are maximized near $\lambda = 0.5$. Thus, the weight $\lambda = 0.5$ is a compromise solution.

DOA. The average ranking performance of each method is shown in Table VII, where we can see that both ORS-OBN and ORS-OTM outperform the benchmark methods, and ORS-OBN is the best one. However, other methods that consider additional information (LDA-F, LDA-L, UCF-F, ICF-F, UIBayes) perform worse than traditional

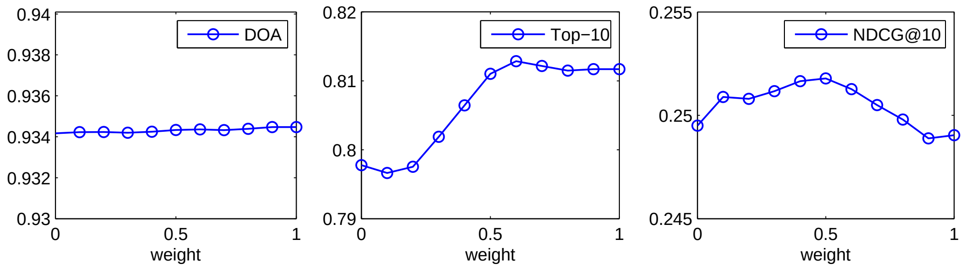
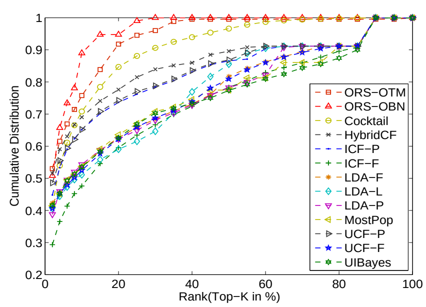


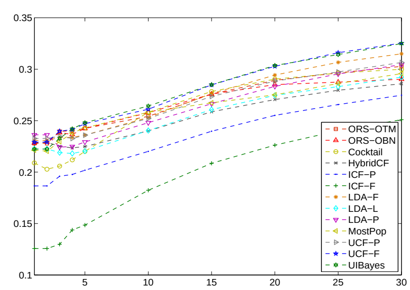
Fig. 6. Results comparison for ORS-OBN with different weights.

Table VII. A performance comparison: DOA(%).

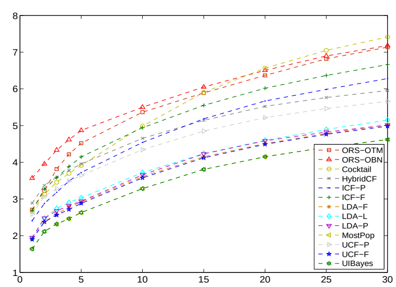
Alg.	UCF-P	ICF-P	UCF-F	ICF-F	LDA-P	LDA-F	LDA-L	HybridCF	MostPop	UIBayes	Cocktail	ORS-OTM	ORS-OBN
DOA(%)	82.54	82.01	76.44	73.69	76.13	76.57	76.91	84.54	75.95	74.88	92.87	93.57	**94.07**



(a) Top-K results



(b) NDCG@k results



(c) Novelty@k results

Fig. 7. Results comparison for different methods.

methods (UCF-P, ICF-P, HybridCF). As we have mentioned previously, properly incorporating additional information into the recommendation model is not a trivial task.

Top-K. In addition, the cumulative distribution of Top-K ranking performances of each method is plotted in Fig. 7(a). As shown in this figure, ORS-OBN still outperforms other methods and the improvement for each K is very significant, and ORS-OTM is the runner-up. The Top-K result is very similar to the DOA result. Note that the result of HybridCF suggests that the hybrid methods based on both items and users get better recommendation quality than the collaborative filterings just based on either of user or item. Please also note that there exists a leap in the lines of some benchmarks,

Table VIII. User study ratings.

Alg.	UCF-P	UCF-F	LDA-F	HybridCF	ORS-OTM	ORS-OBN
Mean	2.86	2.89	2.70	2.90	3.20	**3.26**
Std Dev	0.92	0.99	0.85	0.96	0.88	**0.81**

Table IX. Z-test of user study.

		UCF-P	UCF-F	LDA-F	HybridCF
ORS-OBN	z	3.029	2.640	4.380	2.601
	p	0.998	0.996	0.999	0.995
ORS-OTM	z	2.492	2.142	3.763	2.094
	p	0.994	0.984	0.999	0.982

this is because there are new packages which are not covered by the methods and they are given the same default rank. In summary, these methods focus on mining the relationships between tourists and travel packages, such as ORS-OTM, ORS-OBN, Cocktail, can get better results than other methods.

NDCG. We consider the NDCG scores for different algorithms as shown in Fig. 7(b) with $k = 1, \ldots, 30$. Different from DOA/Top-K, in this metric, ORS doesn't perform the best and item based collaborative filtering methods (i.e., ICF-F, ICF-P) become the worst ones. However, UCF-F performs the best, and UCF-F is the user based collaborative filtering by feature-value pairs. This indicates that even traditional recommendation algorithms can be benefited from the feature-value pair data representation.

Novelty. The evaluation result of the novelty is shown in Fig. 7(c). We note that ORS-OBN, ORS-OTM and Cocktail outperforms other methods because they can consider new package recommendations. Among these three methods, ORS-OBN performs better than Cocktail when K is less than 20 (the top positions in the recommendation list). ORS-OTM also have a better result than Cocktail when K is less than 15. Although Cocktail is better when K is more than 20, it is even worse than ICF-F when K is less than 10, and its overall performance is not good enough. Thus, we can conclude that ORS is more likely to recommend novel packages for each tourist.

User Study. We built a demo system (B/S structure) for making recommendations to end-users (volunteers), and meanwhile collecting their feedbacks by the database (similar to that in [Liu et al.]). When a volunteer enters this system, he/she is first required to report gender and age, and then chooses 5 candidate travel packages that he/she is most possible to buy or already traveled. Based on these information provided by the volunteer, the system outputs and lists the top 50 recommendations of each typical algorithm (i.e., ORS-OTM, ORS-OBN, UCF-P, UCF-F, LDA-F and HybridCF). Next, the volunteer can rate the recommendations on a 5-point Likert scale ranging from 1 (Meaningless) to 5 (Excellent). Here, volunteers conduct "blind reviews" (i.e., they have no idea of each candidate algorithm) and they rate the recommendation results from their own perspectives. Finally, the feedbacks are used to evaluate the performance of each algorithm. In total, we collected 504 ratings for the 6 algorithms (i.e., 84 for each) from 84 volunteers. The final mean ratings and the standard deviations (Std Dev) for each algorithm are shown in Table VIII. We can see that the ratings for ORS-OTM and ORS-OBN are slightly higher than others, and ORS-OBN outperforms ORS-OTM. We also applied z-test as a statistical test, Table IX shows the results. By applying z-test, we find that the differences between the ratings obtained by ORS-OBN and the other four benchmark algorithms are statistically significant with $|z| \geq 2.60$ and $|p| \leq 0.005$. Meanwhile, ORS-OTM is a little bit worse than ORS-OBN when comparing with the other four benchmark algorithms.

6.6. Summary

From the above results, we know that the recommender ORS (including both ORS-OBN and ORS-OTM) performs the best in most situations for travel package recommendation, since it could address the specific challenges existing in the tourism domain. Meanwhile, we could summarize that the two proposed models, OTM and OBN, are effective in the experiments. Furthermore, it suggests that the performance of a recommender system should be evaluated from multiple perspectives, and the choice of a proper system depends on the properties of the specific application [Shani and Gunawardana 2009]. Also, even the additional context information has been effectively handled, e.g., by feature-value pairs, we still need to design the recommendation method carefully for getting better results. For instance, though both UCF-F and ORS-OTM take advantage of the feature-value pair representation, they lead to striking different recommendation results. Thus, it is not a trivial task to effectively aggregate the impact of these additional context information.

7. FEATURE SELECTION USING OTM

Although the ORS framework is more flexible in terms of incorporating additional context information, it should be considered that not all features are helpful enough for the recommendation process. For example, the age and gender of Alice maybe affect her choice, but her height and weight will not have so significant impacts. Meanwhile, in the sampling process of the OTM model, the computation complexity will increase with the feature number growing. It can be concluded that if we develop a feature selection method for ORS, only the useful features will be selected, and better recommendation results may be observed, and meanwhile less computing expense will be cost. As a topic model, OTM has the ability to capture the Information Entropy of Feature-value Pairs as words. Therefore, in this section, we further define the feature information entropy and select features based on the entropy.

7.1. Feature Information Entropy

Information Entropy is a measure of the information content associated with a random variable. In traditional topic models, each word associates with each topic by different probabilities. If we treat the word as the random variable, the word associates a topic as an event, the information entropy of each word w can be defined as follows:

$$E(w) = \sum_{t=1}^{T}(-p(w|T_t)\log p(w|T_t))$$

where T_t means the t-th topic in topic model.

In the OTM model, the features are fixed into the model, and each feature contains some values and each value is treated as a word in traditional topic models. Thus, the information entropy of values also reflects the information content of the associated feature. Specifically, we define the feature information entropy as the average entropy of the associated values:

$$E(f) = \frac{1}{|v \in f|}\sum_{v \in f}\sum_{t=1}^{T}(-p(v|T_t)\log p(v|T_t))$$

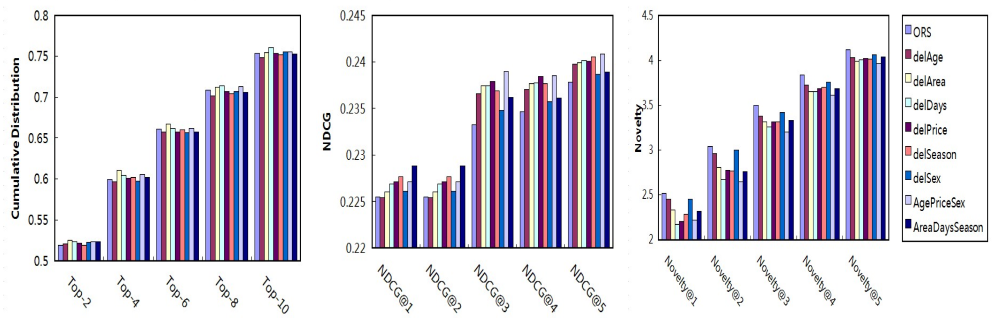
where $v \in f$ means that the feature f contains the value v. In the ORS framework, the information entropies of six features **Age, Gender, Area, Price, Days, Season** can be computed. The results based on our data set are showed in Table X.

Table X. The feature information entropy.

Feature	Area	Age	Days	Gender	Price	Season
Entropy	0.0760	0.0357	0.1474	0.0482	0.0489	0.4699

Table XI. A performance comparison for feature selection: DOA(%).

Alg.	ORS	delAge	delArea	delDays	delGender	delPrice	delSeason	AreaDaysSeason	AgeGenderPrice
DOA(%)	93.57	93.52	93.62	**93.63**	93.58	93.57	93.60	93.54	93.62



(a) Top-K results (b) NDCG@k results (c) Novelty@k results

Fig. 8. Results comparison for feature selection.

7.2. Experimental Evaluation

From the definition of the feature information entropy, it is easy to conclude that higher entropy means the feature has higher randomness related to topics, i.e., cannot distinguish different topics. For this reason, the features with lower entropy should be selected, which can make the ORS framework get better recommendation results and cost less computing expense.

For the purpose of evaluation, we trained six OTM models using corpus which removes one of these six features respectively (denoted as delAge, delArea, delDays, delGender, delPrice, delSeason). We also trained another two OTM models, one using the three features with higher entropies (AreaDaysSeason), and the other choosing the rest three features (AgeGenderPrice). We compared the recommendation results in the ORS framework of all above eight OTM models. For comparison, all the parameters are same with Section 6, and we still adopt DOA, Top-K, NDCG and Novelty as the evaluation metrics.

DOA. The average ranking performance of each method is shown in Table XI, where we can see that the models which remove the features with higher entropy can keep better results and vice versa. If treating ORS as a benchmark, removing the top three features with higher entropy, Area, Days, Season, make the recommendation results better than the benchmark. It supports that selecting the features with lower entropy can get better recommendation results as we previously assumed.

Top-K. In addition, the cumulative distribution of Top-K ranking performances of each method is plotted in Fig. 8(a). Similar to DOA, as shown in this figure, the models which removed the features with higher entropy can still archive better results. It should be noticed that the differentials are relatively small, and for more clear display, we just show the Top-K results from top 2% to top 10%.

NDCG. We consider the NDCG scores for different models as shown in Fig. 8(b) with k = 1, . . . , 5. Same as DOA/Top-K, in this metric, features with lower entropies also make better results, especially for the AgeGenderPrice model, which only considers three features but still get the best result when k = 3, 4, 5. This observation is consistent with our analysis in Section 2, that the features like Age, Gender and Price play important roles when tourists make decisions.

Novelty. At last, the evaluation result of the novelty is shown in Fig. 8(c). Different from other three metrics, removing features make the results becoming worse than ORS, we think the reason is that less features may not be distinguishable enough to find out which package is a novel one.

7.3. Summary for Feature Selection

In this section, we defined feature information entropy for the OTM model and ORS framework, and selected the features based on their entropies. We found that there are some correlations between feature entropy and the recommendation results, and the features with lower entropy actually contain more information and thus lead to better recommendation results. For applying the ORS framework in practice, developers could first use the OTM model for exploiting selecting features, then rebuild the OBN model or OTM model based on the selected features for better recommendations.

8. RELATED WORK

Related work can be grouped into four categories. The first category includes the most relevant work on travel package recommendation. Indeed, researchers have pointed out that some additional context information of travel packages, such as the financial and time cost information, are useful for travel recommendation [Ge et al. 2011]. By considering the travel cost (the financial and time cost), Ge et al. provided a study of cost-aware tour recommendation. Specifically, they developed cost-aware latent factor models, called the GcPMF model, to learn the user/item latent features and user cost preferences simultaneously. However, the GcPMF model is a specific model which only considered the cost-related features. Also, in [Liu et al. 2011], the Tourist-Area-Season Topic (TAST) model was developed. Specifically, Liu et al. noticed that the choices of tourists are related with some features of the travel packages, such as the landscapes, the travel areas, and the travel seasons. Based on these discoveries, they creatively designed the TAST model by considering these three features for travel packages representation. However, the TAST model considered each feature as an independent factor, and it is not very scalable to dynamically incorporate some other useful features.

In the second category, we introduce the related work on intelligent travel self-services[Ricci 2002]. Tourists can use these systems to free plan their tours by themselves, rather than be regulated by the travel service providers. For instance, by exploiting a set of features for each tourist's specific interaction session, Ricci et al. described two case-based reasoning approaches [Ricci et al. 2006a; Ricci et al. 2006b] for travel recommendation and advisory. People also target on providing more context-aware travel information to the on-tour tourists with mobile devices [Ricci 2011; Ricci and Nguyen 2006] which is helpful for exploring the city area. [Mahmood et al. 2009] used conversational systems to autonomously improve the recommendation strategy and applied their approach within a prototype of an online travel recommender system. According to tourists' budgets, [Xie et al. 2010] proposed a composite recommendation method which can give a set of points of interest for travel planning in urban region. Considering that the trip planning is sensitive to the scalability of travel regions, [Lu et al. 2011] proposed a novel data mining-based approach, namely Trip-Mine, for efficient finding of optimal trip within a travel time constraint.

Also, there is a category of research using user location history (recorded in either check-in format or GPS trajectories) to perform travel recommendation in the urban region. By considering the geographic information, [De Carolis et al. 2009] developed a mobile recommender system which helps users make a travel plan in urban region. Also, [Yin et al. 2010] proposed an automatic trip planning framework by leveraging geo-tagged photos and textual travelogues. Moreover, [Hao et al. 2010] proposed a Location-Topic model by learning the local and global topics to mine the location-representative knowledge from a large collection of travelogues, and used this model to recommend the travel destinations. [Zheng et al. 2011] proposed a framework, referred to as a hierarchical-graph-based similarity measurement (HGSM) to uniformly model each individual's location history and effectively measure the similarities among users, in this way both friends and travel locations can be recommended. [Zheng and Xie 2011] modeled multiple users' location histories with a tree-based hierarchical graph (TBHG) and proposed a HITS (Hypertext Induced Topic Search)-based model to infer the interest level of a location and a user's travel experience (knowledge). [Yoon et al. 2012] proposed a social itinerary recommendation by learning from GPS trajectories of both residents and travel experts in a city, which can extract meaningful knowledge about the city. [Wei et al. 2012] also presented a Route Inference framework based on Collective Knowledge to construct the popular routes from uncertain trajectories for helping tourists planning trip routes. With the consideration of both user preferences and social opinions, [Bao et al. 2012] proposed a novel recommender system can facilitate peoples travel not only near their living areas but also to a city that is new to them. [Zheng et al. 2010b; Zheng et al. 2010a; Zheng et al. 2012] developed an integrated and effective mobile recommendation system including three algorithms to answer location-related queries for location-based services. Applying parallel computing technology into recommender systems, [Lu et al. 2012] proposed a novel framework named Personalized Trip Recommendation (PTR) to efficiently recommend the personalized trips meeting multiple constraints of users by mining user's check-in behaviors.

Beyond the tourism domain, the fourth category contains the recommendation works which also exploit additional user/item features. For example, [Basu et al. 1998] applied the inductive rule learner Ripper to the task of recommending movies using both user ratings and content features. Also, [Basilico and Hofmann 2004] designed an SVM-like model with a kernel function that is based on joint features of user ratings as well as attributes of items or users. [Singh and Gordon 2008] provided collective matrix factorization for modeling pairwise relational data, where users' ratings can be encoded using relations of movies, movies' genres, and actors' roles in movies. Moreover, [Bao et al. 2009] proposed a hybrid recommendation system which combines component recommendation engines at runtime based on user/item features. [Cui et al. 2010] built a Feature Interaction Graph (FIG) and employed a probabilistic model based on Markov Random Field to describe the FIG for similarity measure between multimedia Objects. [Agarwal and Chen 2010] proposed fLDA, another matrix factorization method to predict ratings in recommender system applications. Finally, for academic collaboration recommendation, [Tang et al. 2012] proposed the Cross-domain Topic Learning (CTL) model to highlight the existing relationships of authors through implicit topic layers and publications.

However, above methods from other application domains can not be directly applied to tourism domain because of the unique characteristics of the travel data. Meanwhile, the tour recommendation approaches follow a case-by-case manner, and they lack a systematic and open framework to dynamically incorporate multiple types of additional context information. More importantly, to the best of our knowledge, none of existing methods try to provide Object-oriented travel package recommendation.

9. CONCLUSION AND FUTURE WORK

In this paper, we provided *Objected-oriented Recommender System* (ORS) for travel package recommendation. The ORS is an open framework, and has the ability to systematically and cost-effectively incorporate all the available context information.

Specifically, we first analyzed the multiple types of contextual factors from both travel packages and tourists, and these factors can be uniformly represented as feature-value pairs. Then, we proposed the concept of Object, which is the collection of feature-value pairs. Based on the Object-oriented ideas, we proposed two novel models, both of them can extract the implicit relationships among Objects by using the additional context information. By considering the correlation as the latent topics hidden in the collection of feature-value pairs, we first designed an open topic model, *Objected-oriented Topic Model* (OTM) to represent the Objects and identify the tourists' hidden travel interests. For efficiency issue, we then proposed another Bayesian network model, *Objected-oriented Bayesian Network* (OBN) which can quickly infer the co-travel probability of two tourists. Based on the relationships mined by OTM or OBN, the nearest neighbors for each tourist can be found and the recommendation list is generated by the collaborative filtering method. We evaluated the OTM model, the OBN model and the ORS framework on a real-world travel data. The experimental results demonstrated that the ORS framework can lead to better performances for travel package recommendation by incorporating many additional information than several state-of-the-art methods. Finally, we defined feature information entropy for measuring the importance of features, and thus selected the features based on their entropy to achieve the goal of using less features while getting better recommendation results.

Note that there are still many possible directions left for future research. For example, the ORS framework is now mainly focused on travel package recommendation. In the future, we plan to extend it to more general solution for recommendation scenarios in some other application domains.

ACKNOWLEDGMENTS

The work described in this paper was partially supported by grants from National Science Foundation for Distinguished Young Scholars of China (Grant No. 61325010), Natural Science Foundation of China (Grant No. 61073110 and 71329201), Research Fund for the Doctoral Program of Higher Education of China (Grant No. 20113402110024), National Key Technology Research and Development Program of the Ministry of Science and Technology of China (Grant No. 2012BAH17B03). Also, it was supported in part by National Science Foundation (NSF) via grant numbers CCF-1018151 and IIS-1256016.

REFERENCES

ADOMAVICIUS, G. AND TUZHILIN, A. 2005. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *Knowledge and Data Engineering, IEEE Transactions on 17,* 6, 734–749.

AGARWAL, D. AND CHEN, B.-C. 2010. flda: matrix factorization through latent dirichlet allocation. In *Proceedings of the third ACM international conference on Web search and data mining*. ACM, 91–100.

BAO, J., ZHENG, Y., AND MOKBEL, M. F. 2012. Location-based and preference-aware recommendation using sparse geo-social networking data. In *Proceedings of the 20th International Conference on Advances in Geographic Information Systems*. ACM, 199–208.

BAO, T., CAO, H., CHEN, E., TIAN, J., AND XIONG, H. 2010. An unsupervised approach to modeling personalized contexts of mobile users. In *Data Mining (ICDM), 2010 IEEE 10th International Conference on*. IEEE, 38–47.

BAO, X., BERGMAN, L., AND THOMPSON, R. 2009. Stacking recommendation engines with additional meta-features. In *Proceedings of the third ACM conference on Recommender systems*. ACM, 109–116.

BASILICO, J. AND HOFMANN, T. 2004. Unifying collaborative and content-based filtering. In *Proceedings of the twenty-first international conference on Machine learning*. ACM, 9.

BASU, C., HIRSH, H., COHEN, W., ET AL. 1998. Recommendation as classification: Using social and content-based information in recommendation. In *Proceedings of the National Conference on Artificial Intelligence*. John Wiley & Sons LTD, 714–720.

BLEI, D. M., NG, A. Y., AND JORDAN, M. I. 2003. Latent dirichlet allocation. *The Journal of Machine Learning Research 3*, 993–1022.

BREESE, J. S., HECKERMAN, D., AND KADIE, C. 1998. Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence*. UAI'98. Morgan Kaufmann Publishers Inc., 43–52.

CUI, B., TUNG, A. K., ZHANG, C., AND ZHAO, Z. 2010. Multiple feature fusion for social media applications. In *Proceedings of the 2010 international conference on Management of data*. ACM, 435–446.

DE CAROLIS, B., NOVIELLI, N., PLANTAMURA, V. L., AND GENTILE, E. 2009. Generating comparative descriptions of places of interest in the tourism domain. In *Proceedings of the third ACM conference on Recommender systems*. ACM, 277–280.

FAYYAD, U. AND IRANI, K. 1993. Multi-interval discretization of continuous-valued attributes for classification learning. *JPL Technical Report*, 1022–1027.

FOUSS, F., PIROTTE, A., RENDERS, J.-M., AND SAERENS, M. 2007. Random-walk computation of similarities between nodes of a graph with application to collaborative recommendation. *Knowledge and Data Engineering, IEEE Transactions on 19,* 3, 355–369.

GE, Y., LIU, Q., XIONG, H., TUZHILIN, A., AND CHEN, J. 2011. Cost-aware travel tour recommendation. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 983–991.

GRIFFITHS, T. AND STEYVERS, M. 2004. Finding scientific topics. *Proceedings of the National Academy of Sciences of the United States of America 101,* 1, 5228.

HAO, Q., CAI, R., WANG, C., XIAO, R., YANG, J.-M., PANG, Y., AND ZHANG, L. 2010. Equip tourists with knowledge mined from travelogues. In *Proceedings of the 19th international conference on World wide web*. ACM, 401–410.

HARVEY, M., CARMAN, M. J., RUTHVEN, I., AND CRESTANI, F. 2011. Bayesian latent variable models for collaborative item rating prediction. In *Proceedings of the 20th ACM international conference on Information and knowledge management*. CIKM '11. ACM, 699–708.

KOREN, Y. 2008. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 426–434.

LI, Y., LU, L., AND XUEFENG, L. 2005. A hybrid collaborative filtering method for multiple-interests and multiple-content recommendation in e-commerce. *Expert Systems with Applications 28,* 1, 67–77.

LIU, Q., CHEN, E., XIONG, H., DING, C., AND CHEN, J. 2012. Enhancing collaborative filtering by user interest expansion via personalized ranking. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on 42,* 1, 218–233.

LIU, Q., CHEN, E., XIONG, H., GE, Y., LI, Z., AND WU, X. A cocktail approach for travel package recommendation. *Knowledge and Data Engineering, IEEE Transactions on*.

LIU, Q., GE, Y., ZHONGMOU, L., ENHONG, C., AND XIONG, H. 2011. Personalized travel package recommendation. In *Data Mining (ICDM), 2011 IEEE 11th International Conference on*. IEEE, 407–416.

LU, E.-C., LIN, C.-Y., AND TSENG, V. S. 2011. Trip-mine: An efficient trip planning approach with travel time constraints. In *Mobile Data Management (MDM), 2011 12th IEEE International Conference on*. Vol. 1. IEEE, 152–161.

LU, E. H.-C., CHEN, C.-Y., AND TSENG, V. S. 2012. Personalized trip recommendation with multiple constraints by mining user check-in behaviors. In *Proceedings of the 20th International Conference on Advances in Geographic Information Systems*. ACM, 209–218.

MAHMOOD, T., RICCI, F., AND VENTURINI, A. 2009. Improving recommendation effectiveness: Adapting a dialogue strategy in online travel planning. *Information Technology & Tourism 11,* 4, 285–302.

MCCALLUM, A., WANG, X., AND CORRADA-EMMANUEL, A. 2007. Topic and role discovery in social networks with experiments on enron and academic email. *Journal of Artificial Intelligence Research 30,* 1, 249–272.

RESNICK, P., IACOVOU, N., SUCHAK, M., BERGSTROM, P., AND RIEDL, J. 1994. Grouplens: an open architecture for collaborative filtering of netnews. In *Proceedings of the 1994 ACM conference on Computer supported cooperative work*. ACM, 175–186.

RICCI, F. 2002. Travel recommender systems. *IEEE Intelligent Systems 17,* 6, 55–57.

RICCI, F. 2011. Mobile recommender systems. *Information Technology & Tourism 12,* 3, 205–231.

RICCI, F., CAVADA, D., MIRZADEH, N., AND VENTURINI, A. 2006a. Case-based travel recommendations. *Destination Recommendation Systems: Behavioural Foundations and Applications*, 67–93.

RICCI, F., FESENMAIER, D., MIRZADEH, N., RUMETSHOFER, H., SCHAUMLECHNER, E., VENTURINI, A., WÖBER, K., AND ZINS, A. 2006b. Dietorecs: a case-based travel advisory system. *Travel Destination Recommendation Systems: Behavioural Foundations and Applications, CABI Publishing, Wallingford*, 227–239.

RICCI, F. AND NGUYEN, Q. 2006. Mobyrek: A conversational recommender system for on-the-move travelers. *Destination Recommendation Systems: Behavioural Foundations and Applications*, 281–294.

ROSEN-ZVI, M., GRIFFITHS, T., STEYVERS, M., AND SMYTH, P. 2004. The author-topic model for authors and documents. In *Proceedings of the 20th conference on Uncertainty in artificial intelligence*. AUAI Press, 487–494.

SARWAR, B., KARYPIS, G., KONSTAN, J., AND REIDL, J. 2001. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web*. ACM, 285–295.

SHANI, G. AND GUNAWARDANA, A. 2009. Evaluating recommender systems. *Microsoft Research Report MSR-TR-2009-159, Nov*.

SINGH, A. AND GORDON, G. 2008. Relational learning via collective matrix factorization. In *Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 650–658.

TANG, J., WU, S., SUN, J., AND SU, H. 2012. Cross-domain collaboration recommendation. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM.

WEI, L., ZHENG, Y., AND PENG, W. 2012. Constructing popular routes from uncertain trajectories. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 195–203.

XIE, M., LAKSHMANAN, L. V., AND WOOD, P. T. 2010. Breaking out of the box of recommendations: from items to packages. In *Proceedings of the fourth ACM conference on Recommender systems*. ACM, 151–158.

YIN, H., LU, X., WANG, C., YU, N., AND ZHANG, L. 2010. Photo2trip: an interactive trip planning system based on geo-tagged photos. In *Proceedings of the international conference on Multimedia*. ACM, 1579–1582.

YOON, H., ZHENG, Y., XIE, X., AND WOO, W. 2012. Social itinerary recommendation from user-generated digital trails. *Personal and Ubiquitous Computing 16,* 5, 469–484.

YUAN, J., ZHENG, Y., ZHANG, C., XIE, W., XIE, X., SUN, G., AND HUANG, Y. 2010. T-drive: driving directions based on taxi trajectories. In *Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems*. ACM, 99–108.

ZHENG, V. W., CAO, B., ZHENG, Y., XIE, X., AND YANG, Q. 2010a. Collaborative filtering meets mobile recommendation: A user-centered approach. In *Proceedings of the 24rd AAAI Conference on Artificial Intelligence*.

ZHENG, V. W., ZHENG, Y., XIE, X., AND YANG, Q. 2010b. Collaborative location and activity recommendations with gps history data. In *Proceedings of the 19th international conference on World wide web*. ACM, 1029–1038.

ZHENG, V. W., ZHENG, Y., XIE, X., AND YANG, Q. 2012. Towards mobile intelligence: Learning from gps history data for collaborative recommendation. *Artificial Intelligence*.

ZHENG, Y. AND XIE, X. 2011. Learning travel recommendations from user-generated gps traces. *ACM Transactions on Intelligent Systems and Technology (TIST) 2,* 1, 2.

ZHENG, Y., ZHANG, L., MA, Z., XIE, X., AND MA, W. 2011. Recommending friends and locations based on individual location history. *ACM Transactions on the Web (TWEB) 5,* 1, 5.

ZHOU, T., KUSCSIK, Z., LIU, J., MEDO, M., WAKELING, J., AND ZHANG, Y. 2010. Solving the apparent diversity-accuracy dilemma of recommender systems. *Proceedings of the National Academy of Sciences 107,* 10, 4511–4515.