Neurocomputing 254 (2017) 3–12

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

More focus on what you care about: Personalized top reviews set

Wenting Tu [a,*], David W. Cheung [b], Nikos Mamoulis [b]

[a] Department of Computer Science, Shanghai University of Finance and Economics Shanghai, China
[b] Department of Computer Science, The University of Hong Kong, Pokfulam, Hong Kong

ARTICLE INFO

Article history:
Received 28 January 2016
Revised 13 July 2016
Accepted 3 October 2016
Available online 6 March 2017

Keywords:
Personalization
Review ranking
E-commerce
Collaborative filtering

ABSTRACT

Users of e-commerce sites often read reviews of products before deciding to purchase them. Many commercial sites simply select the reviews with the highest quality, according to the votes they have received by users who read the reviews. However, recent work has shown that such a selection may contain redundant information. Therefore, while selecting top reviews, it has been proposed to also consider their coverage (i.e., how many product aspects are covered by them). The goal of this paper is to further improve the top reviews set, using personalization criteria. This is motivated by the fact that the importance of product aspects to different users may vary and users prefer to focus on the most important aspects to them. The objective of our work is to consider the personal preferences of users in review recommendation, by selecting a personalized top reviews set (PTRS), which includes reviews of which the content is related to the aspects important to the user. An experimental evaluation with two public review datasets demonstrates the effectiveness of our approach on computing PTRS that have high quality, coverage, and relevance to the aspects that are important for the user.

1. Introduction

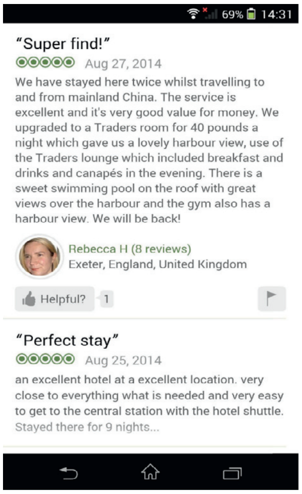
Consumers frequently post their experience with products at e-commerce websites, like Amazon, CNet, and Shopping. These websites allow consumers to post their opinions or reviews and to express preferences or concerns about products. Product reviews are an increasingly important type of user-generated content as they offer a valuable source of information. Product manufacturers and designers, e-commerce websites, and potential consumers can all potentially benefit from the posted data.

Typical online users do not have the patience to go through all reviews of a specific product that interests them [1,2]. This is especially true for mobile applications, where the screen is small and the resources are limited. For example, a 5.0 inch screen typically only fits 2 hotel reviews from TripAdvisor (http://www.tripadvisor.com), as Fig. 3(a) shows. Only the top reviews will be read by most of users and influence users' purchasing decisions. Therefore, selecting and showing only a small *top k reviews set* to a user, from a potentially large number of reviews on the product, is a very important problem. Many online portals solve this problem in a brute-force manner, by allowing users to rate the quality of reviews and selecting the ones with the highest ratings [3]. For example, as Fig. 3(b) presents, by clicking a product to see its reviews in Amazon (http://www.amazon.com), the Amazon website will display the reviews that were highly rated by customers as the most helpful ones. There is also substantial amount of research on automatically estimating the quality of a review [4–8]. Since these approaches do not account for the redundancy in the content of the reviews, some important aspects of the reviewed item may not be covered at all by the top reviews. To make the set of top reviews cover as many as possible different viewpoints of the product (i.e., product aspects), some recent work [1,3,9] also takes the *coverage* of product aspects into consideration as well as the overall quality of the top reviews set.

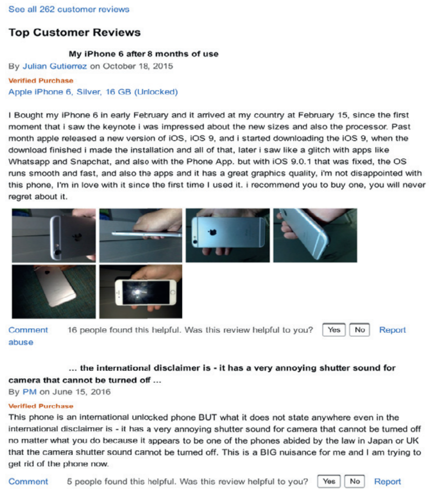
To the best of our knowledge, although there are some previous works that noticed the fact that the various product aspects may have different importance to different users [10–12], none of them make use of this fact for improving review ranking. As a result, existing methods show the same set of top reviews to different users (e.g., u_i and u_j), regardless of any differences in their interests. The goal of this paper is to improve the quality of the top reviews shown to a user when he/she is investigating some product, by predicting which product aspects are most important to her. Compared with previous algorithms, our method selects reviews that not only are of high quality and cover many aspects, but also focus more on the product aspects that are important to the user.

* Corresponding author.
E-mail address: w.tingtu@gmail.com (W. Tu).

http://dx.doi.org/10.1016/j.neucom.2016.10.081



(a) Tripadvisor reviews on a 5.0-inch screen.



(b) Amazon reviews.

Fig. 1 illustrates the applicability of our approach by a concrete example. Assume that users *ronischuetz* and *Rebecca* have visited *Hotel Jen* and left their comments about the hotel. In his comment, *ronischuetz* mentions aspects *location, staff, pool* and *view*. It is reasonable to infer that these aspects of *Hotel Jen* are important to *ronischuetz*. Meanwhile, *Rebecca* discusses different aspects in her review, including *price, breakfast, pool* and *view* of *Hotel Jen*. Our objective is that, when *ronischuetz* (*Rebecca*) first browses the reviews of Hotel Jen (which means that they have not left their comments about Hotel Jen yet), we could show him (her) the reviews that not only are of high quality and cover many aspects, but also provide rich information about *location, staff, pool* and *view* (*price, breakfast, pool* and *view*) to *ronischuetz* (*Rebecca*), as shown in the right part of Fig. 1.

To realize the above goal, we need to address two issues. First, note that when a user first browses the reviews of Hotel Jen, we do not have her comments about the hotel. Therefore, we cannot directly know which aspects are important to the user. Therefore, the first issue is to *predict* which aspects are important to a user for a specific item before the user reviews the item. After identifying which aspects of Hotel Jen are important to the user, the second question is how to select the reviews to present to her, considering the product aspects important to her. Accordingly, our methodology for retrieving the *personalized top reviews set* (PTRS) for an item consists of two steps. The first is to predict which product aspects are important to a the target user. The second is to weigh the selection in order to favor reviews that discuss these aspects.

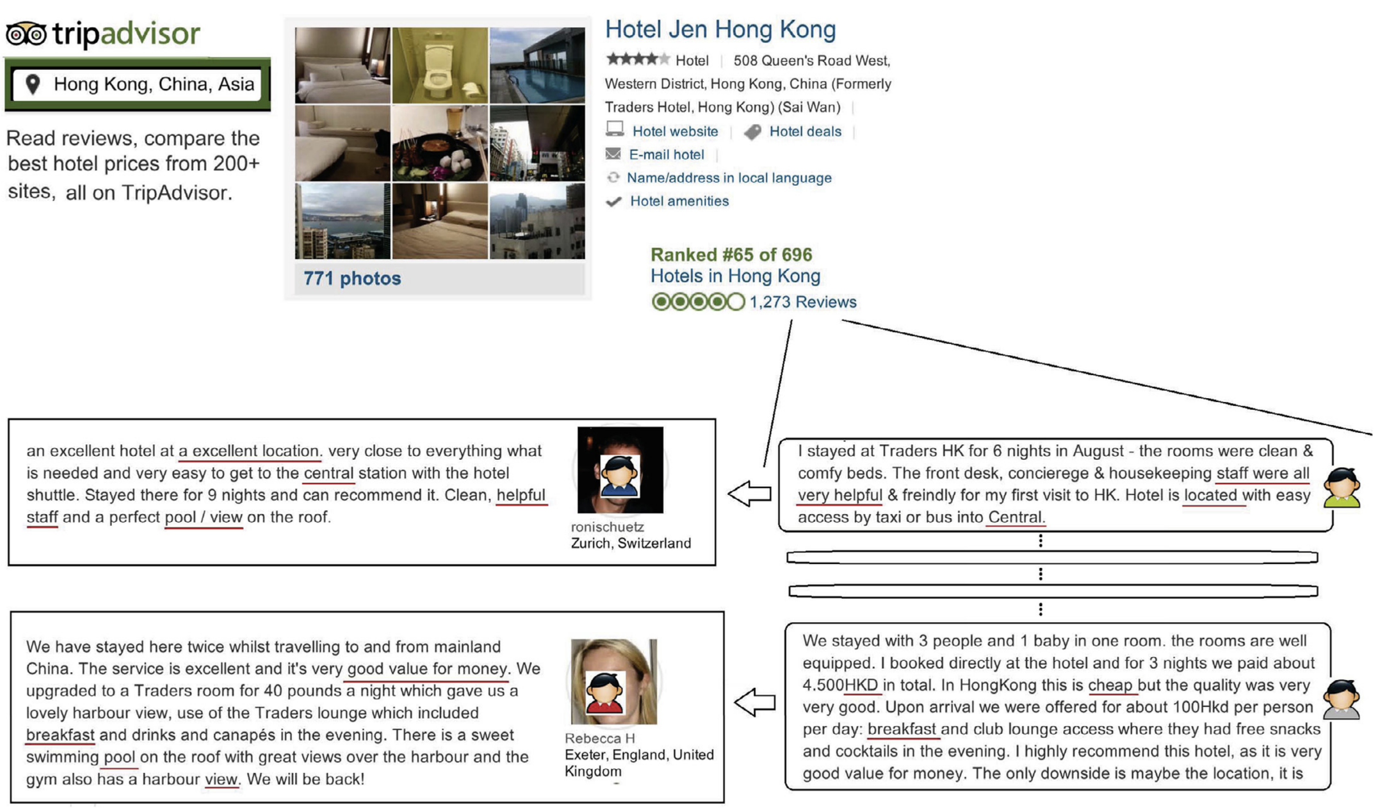


Fig. 1. Graphical illustration of the motivation of our work.

While the selected reviews focus more on these important aspects, at the same time, we also consider their coverage and quality.

We conducted experiments using two real-world datasets to compare the top review sets obtained by our methodology and alternative approaches. In our comparison, we test whether the qualities of the selected reviews are high, whether the reviews can cover as many product aspects as possible, and whether the user can view the reviews that refer to the product aspects important to her earlier. The results demonstrate the superiority of our approach in the above aspects.

In Section 2, we define some notations and formulate the top-reviews set selection problem. In Section 3, we introduce the selection model of Tsaparas et al. [1], which we improve. Section 4 presents our methodology and Section 5.1 presents our experimental analysis. Related work is briefly reviewed in Section 7. Finally, we conclude and give directions for future work in Section 8.

2. Problem formulation

2.1. Notations

We first define some notations. We assume a set $\mathcal{U} = \{u_1, u_2, \dots, u_x\}$ of users, and a set $\mathcal{P} = \{p_1, p_2, \dots, p_n\}$ of products. All products in $\mathcal{P}$ belong to same domain (e.g., hotel or restaurant) and share a set of product aspects $\mathcal{A} = \{a_1, a_2, \dots, a_m\}$. For example, $\mathcal{U}$ includes the users on Tripadvisor website, $\mathcal{P}$ includes registered hotels on the site, while $\mathcal{A}$ includes “price”, “cleanliness”, “business service”, “location” and other common attributes of hotels. Users can write reviews to products; We use r_i^j to denote the review written by u_i to describe his/her opinion about product p_j. Each review includes text and a rating (e.g., number of stars given by u_i to p_j).

2.2. Ground truth of “What you will care about a product”

Suppose our objective is to show to user u_i a set of reviews written for describing product p_j. In this section, we discuss about how could we know the ground truth of “What u_i will care about the product p_j” (i.e., what aspects of p_j are important to u_i). We called this knowledge as *aspect-importance distribution*. The *aspect-importance distribution* of user $u_i \in \mathcal{U}$ to product $p_j \in \mathcal{P}$ is a m-dimensional vector $\Phi_j^i = (\phi_1^{i,j}, \phi_2^{i,j}, \dots, \phi_m^{i,j})$. Based on the aspect set $\mathcal{A} = \{a_1, a_2, \dots, a_m\}$, each value $\phi_l^{i,j}$ of Φ_j^i corresponds to the importance level of aspect a_l of product p_j to user u_i. In our work, we extract the ground truth of *aspect-importance distribution* Φ_j^i from the content of review r_i^j. Specifically, if user u_i has reviewed product p_j, then Φ_j^i can be generated from the review. For this, we assume that when a user reviews a product, she will use more words to describe the aspects that are more important to her. Thus, we model the importance of aspect a_l for u_i on product p_j as the percentage of words in the review r_j^i used for describing a_l. Let us take the review written by *ronischuets* (as shown in Fig. 1) as an example. *ronischuets* wrote the sentence “An excellent hotel at an excellent location, very close to everything what is needed and very easy to get to the central station with the hotel shuttle.” for describing aspect *location*, which indicates that *location* of Hotel Jen plays important role to *ronischuets*’s feeling about Hotel Jen (i.e., *location* of Hotel Jen is important to *ronischuets*). Meanwhile, *ronischuets* also mentioned aspects *staff, pool* and *view* in the sentence “Helpful staff and a perfect pool/view on the roof.”. Note that this sentence is much shorter than the sentence talking about *location*, which indicates that *ronischuets* used much less words to describe *staff, pool* and *view*. Thus, the importance of *staff, pool* and *view* should be much less than the importance of *location*.

According to the above idea, for each aspect a_l, we measure in a quantity $\hat{\phi}_l^{i,j}$ the total number of words in the sentences in r_j^i that mention a_l. For example, if we regard *ronischuets* as u_i and Hotel Jen as p_j, we use the number of words that are related to location aspect a_l in sentence “An excellent hotel at a excellent location, very close to everything what is needed and very easy to get to the central station with the hotel shuttle” to be the indicator for $\hat{\phi}_l^{i,j}$. However, note that this is only reasonable for sentences talking about only one aspect since, if a sentence mentions one aspect only, all words in it should be related to that aspect. Thus, if a sentence talks about multiple aspects, we assume that the words equally refer to all the mentioned aspects. For example, in sentence “Helpful staff and a perfect pool/view on the roof”, we assume the one third of words are used for describing *staff*, one third for *pool* and one third for *view*.

According to the above discussion, we now formally define $\hat{\phi}_l^{i,j}$. Assuming that $\mathcal{S}_l^{i,j}$ indicates the sentences contained in r_j^i and mention aspect a_l, each $\phi_l^{i,j}$ $(l = 1, 2, \dots, m)$ can be defined as follows:

$$\hat{\phi}_l^{i,j} = \sum_{s \in \mathcal{S}_l^{i,j}} \frac{\mathcal{N}^w(s)}{\mathcal{N}^a(s)}, \quad \phi_l^{i,j} = \frac{\hat{\phi}_l^{i,j}}{\sum\limits_{a_{l'} \in \mathcal{A}} \hat{\phi}_{l'}^{i,j}}. \tag{1}$$

Here, $\hat{\phi}_l^{i,j}$ actually equals to the number of words used for describing aspect a_l in review r_j^i, where $\mathcal{N}^w(s)$ denotes the number of words contained in the sentence s and $\mathcal{N}^a(s)$ is the number of aspects mentioned in s. If the sentence s mentions aspect a only, we assume that u_i uses all $\mathcal{N}^w(s)$ words for describing the aspect a of p_j. However, if there are more than one aspects mentioned in s, for each single aspect, the words used for describing it is close to $\mathcal{N}^w(s)$ divided by $\mathcal{N}^a(s)$. Finally, we normalize $\hat{\phi}_l^{i,j}$ $(l = 1, 2, \dots, m)$ so that the final $\phi_l^{i,j}$ $(l = 1, 2, \dots, m)$ sum to unity. With this formula, the *aspect-importance distribution* from *ronischuets* to Hotel Jen will be [0.75(location), 0.083(staff), 0.083(view), 0.083(pool)] (we do not show the aspects that got zero importance).

2.3. Personalized Top Reviews Set (PTRS) selection problem

We denote the set of reviews by user u_i as $\mathcal{R}_i^u$ and the set of reviews written for product p_j as $\mathcal{R}_j^p$. The Personalized Top Reviews Set (PTRS) selection problem can be defined as follows: *given a user u_i, a product p_j, and a user-defined integer k, retrieve a k-sized subset $\hat{\mathcal{R}}_{i,j}$ of $\mathcal{R}_j^p$ to show to user u_i.* The criteria for selecting the top reviews set in previous work are the average quality (votes) of reviews contained in $\hat{\mathcal{R}}_{i,j}$ and the number of p_j’s aspects covered by $\hat{\mathcal{R}}_{i,j}$. Since previous work only relies on quality of reviews and coverage metrics, the generated top review sets for a product are the same for different users (i.e., $\hat{\mathcal{R}}_{x,j} = \hat{\mathcal{R}}_{y,j}$ for any $u_x, u_y \in \mathcal{U}, x \neq y$). In our work, we first predict Φ_j^i, which is sensitive to the user u_i, and then consider it when we retrieve TRS. Thus, our method may generate different top review sets for different users (i.e., we may have $\hat{\mathcal{R}}_{x,j} \neq \hat{\mathcal{R}}_{y,j}$ for two different users u_x and u_y). In summary, the objective of the Personalized Top Reviews Set (PTRS) selection problem is that the selected TRS should not only be of high-quality and high-coverage, but should also weigh on the different importance of the various product aspects to users.

3. Preliminary: comprehensive TRS (CTRS) selection

Tsaparas et al.[1] noted that simply ranking the reviews based on their ratings by users and selecting the top-k ones, may result in redundancy in the content of the reviews in the TRS; therefore they propose a methodology to retrieve a *comprehensive* top reviews set (CTRS). Comprehensiveness is defined with respect to the aspects of the product and the qualities of the reviews: the selected $\hat{\mathcal{R}}_{i,j}$ should cover as many aspects related to product p_j as possible; at the same time, the review quality is also considered in the selection. We now discuss the details of selecting a CTRS: we assume a quality function q that maps a review r to a real number $q(r)$, measuring e.g. the helpfulness of the review. This value may be derived from the data (for example, by averaging the votes that users gave to r), or it may be inferred algorithmically. The objective of CTRS selection is to include reviews that have high quality, while covering as many different aspects of p as possible. For this purpose, the scoring function of a TRS candidate $\hat{\mathcal{R}}$ is defined as:

$$F(\hat{\mathcal{R}}) = \sum_{a_l \in \mathcal{A}} f(\hat{\mathcal{R}}, a_l), \tag{2}$$

where $f(\hat{\mathcal{R}}, a_l)$ indicates how much $\hat{\mathcal{R}}$ contributes on providing information on aspect a_l of the product, defined as:

$$f(\hat{\mathcal{R}}, a_l) = max_{r \in \hat{\mathcal{R}}_{a_l}} q(r), \tag{3}$$

where $\hat{\mathcal{R}}_{a_l}$ denotes the set of reviews in $\hat{\mathcal{R}}$ that mentioned a_l. Then, to find the TRS that maximizes the scoring function, Tsaparas et al. [1] used an easy-to-implement greedy algorithm (Algorithm 1).

Algorithm 1 CTRS selection.

Input:
Set of reviews on the target product p_j: $\mathcal{R}_j^p$;
Set of product aspects: $\mathcal{A}$;
Integer budget value: k;
Scoring function: F;
Output:
Top review set: $\hat{\mathcal{R}}_{i,j}$

```
R̂^0_{i,j} = ∅;
for x = 1, ..., k do
    for all r ∈ R_p \ R̂^{x-1}_{i,j} do
        △_{x-1}(r) = F(R̂^{x-1}_{i,j} ∪ {r}) − F(R̂^{x-1}_{i,j});
    end for
    r_x = arg max_{r ∈ R_p \ R̂^{x-1}_{i,j}} △_{x-1}(r) ;
    R̂^x_{i,j} = R̂^{x-1}_{i,j} ∪ {r};
end for
return R̂^k_{i,j};
```

The algorithm performs k iterations, incrementally building the review set, by adding one review at the time. Specifically, in each iteration, it selects the review that achieves the maximum incremental gain $\triangle_{x-1}(r)$ in the cumulative score function F. According to the definition of F (see Eqs. (2) and (3)), high quality review which provides information for aspects have not been covered yet will be selected each time. In the end, the top review set $\hat{\mathcal{R}}_{i,j}^k$ computed by the CTRS greedy algorithm will contain reviews not only of high quality but also of high coverage with respect to the product aspects.

4. Our work: personalized TRS selection

In order to further improve the quality of $\hat{\mathcal{R}}_{i,j}$, in this section, we discuss how to select a set of reviews about p_j which not only contain high-quality reviews and cover multiple aspects, but also weigh on the different importance of the various product aspects to the target user.

4.1. Predicting personalized aspect-importance distributions

Note that when a user (e.g., u_i) searches information about a product (e.g., p_j) for the first time, we should have not yet obtained a review written by u_i on p_j (i.e., r_j^i). This means that we cannot compute the real *aspect-importance distribution* Φ_j^i from r_j^i. Therefore, in order to retrieve a $\hat{\mathcal{R}}_{i,j}$ based on Φ_j^i, we should first predict Φ_j^i.

Here, we solve the problem of predicting Φ_j^i by applying the idea of user-based collaborative filtering [13–15]: aggregating preferences from u_i's similar users to target item to infer the preference from u_i to the target item. Specifically, to predict the *aspect-importance distribution* Φ_j^i from our target user u_i to p_j, we aggregate all *aspect-importance distribution* Φ_j^v, for each user u_v who has commented on p_j; in the aggregation, each Φ_j^v is weighed based on the similarity between u_v and u_i with respect to their aspect tastes.

For this, we first need a way to evaluate similarities between users' aspect tastes. A nature idea is to use the average of *aspect-importance distributions* revealed in u_i's historical reviews as u_i's *aspect profile* (denoted by λ^i). This assumes that two users are similar if they always agree with each other about which aspects of products in $\mathcal{P}$ are important in the past. Specifically, we have:

$$w^{i,v} = \cos(\lambda^i, \lambda^v), \tag{4}$$

$$\lambda^i = \frac{1}{|P^i|} \sum_{p'_j \in P^i} \Phi_{j'}^i, \qquad \lambda^v = \frac{1}{|P^v|} \sum_{p'_j \in P^v} \Phi_{j'}^v, \tag{5}$$

where $w^{i,\,v}$ is the user similarity between u_i and u_v, which is defined as cosine similarity between the aspect profiles λ^i and λ^v (i.e., means of u_i and u_v's historical *aspect-importance distributions*).

Then we predict Φ_j^i (denoted as $\tilde{\Phi}_j^i$) as:

$$\tilde{\Phi}_j^i = \frac{\sum_{u_v \in U_j} w^{i,v} \Phi_j^v}{\sum_{u_y \in U_j} w^{i,y}}, \tag{6}$$

where U_j includes the users who have reviewed p_j.

4.2. Personalized TRS selection

After obtaining the estimated *aspect-importance distribution* $\tilde{\Phi}_j^i = (\tilde{\phi}_1^{i,j}, \tilde{\phi}_2^{i,j}, \ldots, \tilde{\phi}_m^{i,j})$ for (u_i, p_j) pair, the next step is to integrate it into the procedure of retrieving $\hat{\mathcal{R}}_{i,j}$, in order for retrieving a *personalized* TRS that focuses more on the product aspects that are important to the user. Recall that in the CTRS selection process, introduced in Section 3, the scoring function $F(\hat{\mathcal{R}})$ gives equal weights to all aspects a, when aggregating their $f(\hat{\mathcal{R}}, a)$. Thus, CTRS regards that all aspects have equal and static (i.e., user-independent) importance to all users. Our proposal is to consider the user's aspect-importance distribution in CTRS, by adjusting the weight of each product aspect in the scoring function $F(\hat{\mathcal{R}})$, accordingly. Specifically, assuming that we are retrieving the TRS among reviews on product p_j to user u_i, we define the *personalized* scoring function $F_p(\hat{\mathcal{R}})$ of a TRS candidate $\hat{\mathcal{R}}$ as follows:

$$F_p(\hat{\mathcal{R}}) = \sum_{a_l \in \mathcal{A}} (\tilde{\phi}_l^{i,j} + \delta) f(\hat{\mathcal{R}}, a_l), \tag{7}$$

where $\tilde{\phi}_l^{i,j}$ is the predicted importance of aspect a_l of p_j to user u_i and δ is a normalization parameter that takes a very small value. For example, in our experiments, we found that more than 95% non-negative aspect importance values are larger than 10^{-4}. Therefore, when retrieving the personalized TRS, we set δ as 10^{-4} in

Table 1
Extracted product aspects.

Dataset	Product aspects
TripAdvisor (hotel review)	Price, cleanliness, business service, balcony, food, location, bed, air, sleep quality, tv, view, size, room, bathroom, service, pool
Yelp (restaurant review)	Discount, atmosphere, cleanness, service, attitude, efficient, professional, waiter, reservation, salad, burger, cheese, fried, chinese, fresh, beans, sushi, meat, soup, rice, mexican, veggies, sauce, tasty, sandwich, pizza, beef, desserts, grab, mojito, alcoholic, appetizers, refills, tea, drinks, rum, refill, martini, juice, drink, mixed, iced, margarita, water, boba, espresso, coffee, bartenders, soda, vodka, bartender, alcohol, fountain

order for δ not to dominate the impact of predicted aspect importance information. By adding δ to $\tilde{\phi}_l^{i,j}(l = \{1, \ldots, m\})$, the weights of the product aspects whose importance values in our prediction are zero become non-zero, giving the reviews referring those product aspects chances to be selected. This way, the selection results are less sensitive to data sparsity (i.e., when the reviews written by the user are very few and some product aspects that the user cares about are not mentioned).

Algorithm 2 shows how the personalized TRS for (u_i, p_j) is selected.

Algorithm 2 Personalized TRS selection.

Input:
Set of reviews on the target product p_j: $\mathcal{R}_j^p$;
Set of users have commented on p_j: U_j
Set of product aspects: $\mathcal{A}$;
Integer budget value: k;
Scoring function: F;
Output:
Top review set: $\hat{\mathcal{R}}_{i,j}$
Calculate λ^i and λ^v for each $u_v \in U_j$;
Calculate $w^{i,v}$ for each $u_v \in U_j$;
Calculate $\tilde{\Phi}^{i,j}$ by Eq. (6);
$\tilde{\mathcal{R}}^0 = \emptyset$;
for all $i = 1, \ldots, k$ **do**
  for all $r \in \mathcal{R}_j \setminus \tilde{\mathcal{R}}^{i-1}$ **do**
    $\triangle_{i-1}(r) = F_p(\tilde{\mathcal{R}}^{i-1} \bigcup \{r\}) - F_p(\tilde{\mathcal{R}}^{i-1})$;
  end for
  $r_i = \arg\max_{r \in \mathcal{R}_p \setminus \tilde{\mathcal{R}}^{i-1}} \triangle_{i-1}(r)$;
  $\tilde{\mathcal{R}}^i = \tilde{\mathcal{R}}^{i-1} \bigcup \{r\}$;
end for
Set $\tilde{\mathcal{R}}^k$ as the top k reviews showing to u_i

5. Experimental evaluation

5.1. Data preparation and experiments setup

We conduct our experiments on two real public datasets that have also been used in previous research in review mining. The first consists of 878,561 reviews on 4333 hotels crawled from TripAdvisor [16]. The second contains 158,430 reviews on 4503 restaurants from Yelp [17]. For each review, we have the review content, the rating of the review on the product, and the votes by other users on how helpful the review is. For each review r, we use the number of positive votes as a measure of the review's quality $q(r)$. For identifying product aspects, we use the methodology proposed in [18] and obtain the product aspects for TripAdvisor's hotel reviews and Yelp's restaurant reviews, as shown in Table 1.

In the experiments, from each review on a product p_j by a user u_i, we extract the real aspect-importance distribution Φ_j^i as the ground truth for the pair (u_i, p_j). In order to predict $\tilde{\Phi}_j^i$, we remove the review from the dataset and use our methodology described in Section 4.1. We compare our prediction with the ground truth in order to test the accuracy of our approach. Then, we use our PTRS selection approach[1] (Section 4.2) to retrieve the set $\hat{\mathcal{R}}_{i,j}$ of reviews recommended to u_i for product p_j.

In all cases, we set $k \in \{1, 2, 5\}$, considering that only a few reviews could be regraded as top reviews, especially in a mobile application, where we do not expect the user to be able to read more than five reviews. For personalized TRS selection, we set $\delta = 10^{-4}$ in Eq. (7).

5.2. Experimental results and analysis

Recall that our method includes two steps. The first is to predict $\tilde{\Phi}_j^i$, and the second is to retrieve personalized top review set $\hat{\mathcal{R}}_{i,j}$ by taking $\tilde{\Phi}_j^i$ into consideration.

5.2.1. Comparison on predicting aspect-importance distributions

Before comparing the top review sets obtained by our method and other competitors, we first evaluate whether our method discussed in Section 4.1 could predict the *aspect-importance distribution* effectively. To evaluate the prediction accuracy, we use cosine, Pearson and Spearman correlations to measure of how close our prediction $\tilde{\Phi}_j^i$ is to the ground truth Φ_j^i. The average of correlation values on all (u_i, p_j) pairs is used for evaluating prediction accuracy.

As introduced in Section 4.1, our method uses the user-based collaborative filtering framework to predict the aspect-importance distribution. In our approach, we build λ^i (i.e. the aspect profile of user u_i) as the mean of *aspect-importance distributions* in the past reviews of u_i. We denote this approach for building aspect profiles as Past Aspect Importance Distribution (PAID) based strategy. Note that some of previous work (which are proposed for recommendation and review summarization) can potentially be used to construct user aspect profiles since their algorithms also contain the step to identify important product aspects [10–12]. These works predict which product aspects are important to users by mainly relying on two assumptions: (a) important aspects are frequently commented in consumer reviews; and (b) consumers' opinions on these aspects greatly influence their overall rating on the product. Briefly speaking, according to the first assumption, the importance of an aspect in a user's aspect profile will be proportional to the times that the aspect keywords appear in his/her past reviews. We call this kind of aspect profile as *frequency based aspect profile*. Otherwise, according to the second assumption, the importance of a aspect in a user's aspect profile is modeled as the aspect's influence to the overall ratings given by the user. We call the strategies based on these two assumptions *frequency based* and *rating based*, respectively. Finally, we will compare the performance of using the user-based collaborative filtering framework with aspect profiles obtained by the PAID based method, the frequency based method and the rating based method. We will also show the performances when we directly use aspect profiles of users as their aspect-importance distribution to see whether using the

[1] We used Python to implement our algorithm and other competitors. The source code will be publicly available when this paper is published.

Table 2
Performance of predicting personalized aspect-importance distributions.

Method	On Tripadvisor			On Yelp		
	Cosine	Pearson	Spearman	Cosine	Pearson	Spearman
Frequency based	0.48	0.37	0.43	0.25	0.19	0.18
Rating based	0.45	0.36	0.43	0.24	0.20	0.21
PAID based	0.60	0.46	0.45	0.29	0.22	0.23
Frequency + UCF	0.67	0.59	0.54	0.39	0.35	0.33
Rating + UCF	0.61	0.50	0.49	0.37	0.33	0.32
PAID + UCF	0.71	0.60	0.59	0.44	0.40	0.41

user-based collaborative filtering framework is helpful in predicting aspect-importance distribution.

Table 2 shows the results on Yelp and Tripadvisor datasets, respectively. "Frequency based", "Rating based", and "PAID based" correspond to the user aspect profiles obtained by frequency based, rating based, and PAID based methods, respectively, as aspect-importance distributions directly. "Frequency + UCF", "Rating + UCF" and "PAID + UCF" indicate using aspect profiles obtained by the three approaches as aspect-importance distributions as user aspect profiles in the user-based collaborative filtering framework. We can see that the aspect-importance distributions predicted by "PAID + UCF" are the most positively correlated to the real aspect-importance distributions. This indicates that the approach (i.e. "PAID + UCF") is the most effective method for predicting aspect-importance distributions.

5.2.2. Comparison on selecting top review set

Next, we compare our personalized TRS approach to CTRS, i.e., the set of reviews computed by the methodology of Tsaparas et al. [1]. Moreover, we also use the quality of reviews as the only selection criterion for TRS selection and obtain another (baseline) competitor, denoted by QTRS (i.e., Quality-based Top Reviews Set). For evaluating a TRS, we measure (i), whether the qualities of the reviews in TRS are high, (ii), whether the reviews cover as many product aspects as possible and (iii), whether the reviews match the *aspect-importance distribution* of the target user. In other words, whether more content in the reviews is about the aspects important to the target user.

More specifically, for each (u_i, p_j) pair, We obtain the personalized top review set (PTRS) $\hat{\mathcal{R}}^{pers}_{i,j}$, based on our methodology (Algorithm 2), the comprehensive top-review-set (CTRS) $\hat{\mathcal{R}}^{comp}_{i,j}$ (Algorithm 1), and a quality-based top-review-set (QTRS) $\hat{\mathcal{R}}^{qual}_{i,j}$ by sorting the $\mathcal{R}_j$ according to $q(\cdot)$. Then, we compare $\hat{\mathcal{R}}^{pers}_{i,j}$, $\hat{\mathcal{R}}^{comp}_{i,j}$ and $\hat{\mathcal{R}}^{qual}_{i,j}$ using the following measures:

- The *quality* (Qua) of the reviews in a TRS is measured by dividing the average quality of the reviews in the TRS by the maximum quality of all reviews in $\mathcal{R}_j$ (for scaling):

$$Qua_{i,j} = \frac{1}{|\hat{\mathcal{R}}_{i,j}|} \sum_{r \in \hat{\mathcal{R}}_{i,j}} \frac{q(r)}{\max_{r' \in \mathcal{R}_j} q(r')}. \tag{8}$$

- The *coverage* (Cov) of a TRS is measured by the number of product aspects covered by the reviews in it, divided by the total number of aspects ($|\mathcal{A}|$):

$$Cov_{i,j} = \frac{1}{|\hat{\mathcal{R}}_{i,j}|} \sum_{r \in \hat{\mathcal{R}}_{i,j}} \frac{A(r)}{|\mathcal{A}|}, \tag{9}$$

where $A(r_j)$ indicates the set of aspects covered by r.

- The *personalized match* (Pmat) measures whether the reviews provide more information about the aspects important to users. Recall that the aspect importance distribution extracted from a review actually reveals how much information is used to describe different aspects (see Eq. (6) for details). Therefore, whether a review on the target product p_j provides more information about the aspects to the target user u_i could be evaluated by the correlation between the review's aspect importance distribution and the real aspect importance distribution Φ^i_j of u_i on p_j, i.e., the ground truth. Subsequently, the Pmat of review set $\hat{\mathcal{R}}_{i,j}$ is the average of the correlation values between Φ_r, for each $r \in \hat{\mathcal{R}}_{i,j}$, and Φ^i_j:

$$Pmat_{i,j} = \frac{1}{|\hat{\mathcal{R}}_{i,j}|} \sum_{r \in \hat{\mathcal{R}}_{i,j}} Cor(\Phi^i_j, \Phi_r), \tag{10}$$

where Φ_r is the aspect importance distribution of review r and $Cor(\Phi^i_j, \Phi_r)$ is defined as the Cosine, Pearson, or Spearman correlation (we will show evaluation results for each of the three correlation metrics).

Fig. 2 shows the average goodness of PTRS $\hat{\mathcal{R}}^{pers}_{i,j}$, CTRS $\hat{\mathcal{R}}^{comp}_{i,j}$ and QTRS $\hat{\mathcal{R}}^{qual}_{i,j}$ on all (user, product) pairs, for each of the three measures (i.e., Qua, Cov, and Pmat). As expected, the average qualities of reviews in QTRS are higher than the ones in PTRS and CTRS, since quality is the only selection criterion used in QTRS. However, the coverage of QTRS is lower compared to those of PTRS and CTRS. This is consistent to the observations in [1]: ranking reviews based on their quality only results in redundancy in the content of the TRS. By comparing the performance of PTRS and CTRS, we can see that PTRS obtains much higher Pmat, which indicates PTRS provides richer information about the product aspects that are important to the users. Meanwhile, compared with CTRS, PTRS achieves only marginally lower coverage. Summing up, in the TRS obtained by our method, the users not only see high-quality reviews covering many product aspects, but also find information mostly related to the product aspects that are important to them.

Parameter-sensitive Analysis: Fig. 3 shows a parameter sensitivity analysis on how δ influences PTRS performance. We vary δ values in $\{10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1.0\}$ and present Pmat scores (i.e., the average of Cosine, Pearson, and Spearman correlation scores) corresponding to different δ values. As the figure shows, when δ is small enough, PTRS performs stably. When δ is larger than 10^{-4}, the performance will decrease sharply with increasing δ. This is because the influence of aspect-importance distribution in the scoring function reduces when δ increases. Thus, we suggest setting δ to a sufficiently small value (e.g., smaller than 95% of the non-negative aspect importance values) in practice.

Case study: we extracted a test case from the experimental results on the Tripadvisor dataset, in order to illustrate in a more intuitive way the characteristics of top reviews sets obtained by the three approaches. Specifically, the first row of Table 3, presents the review of a user u_i to a product p_j. From the review content, we can see that u_i mentions aspects *price, room, food* and *location* in r^i_j, indicating that these aspects of p_j are the most important to her. In the next rows of the table, we show the top-1 review in $\hat{\mathcal{R}}^{pers}_{i,j}$, $\hat{\mathcal{R}}^{comp}_{i,j}$ and $\hat{\mathcal{R}}^{qual}_{i,j}$. We observe that:

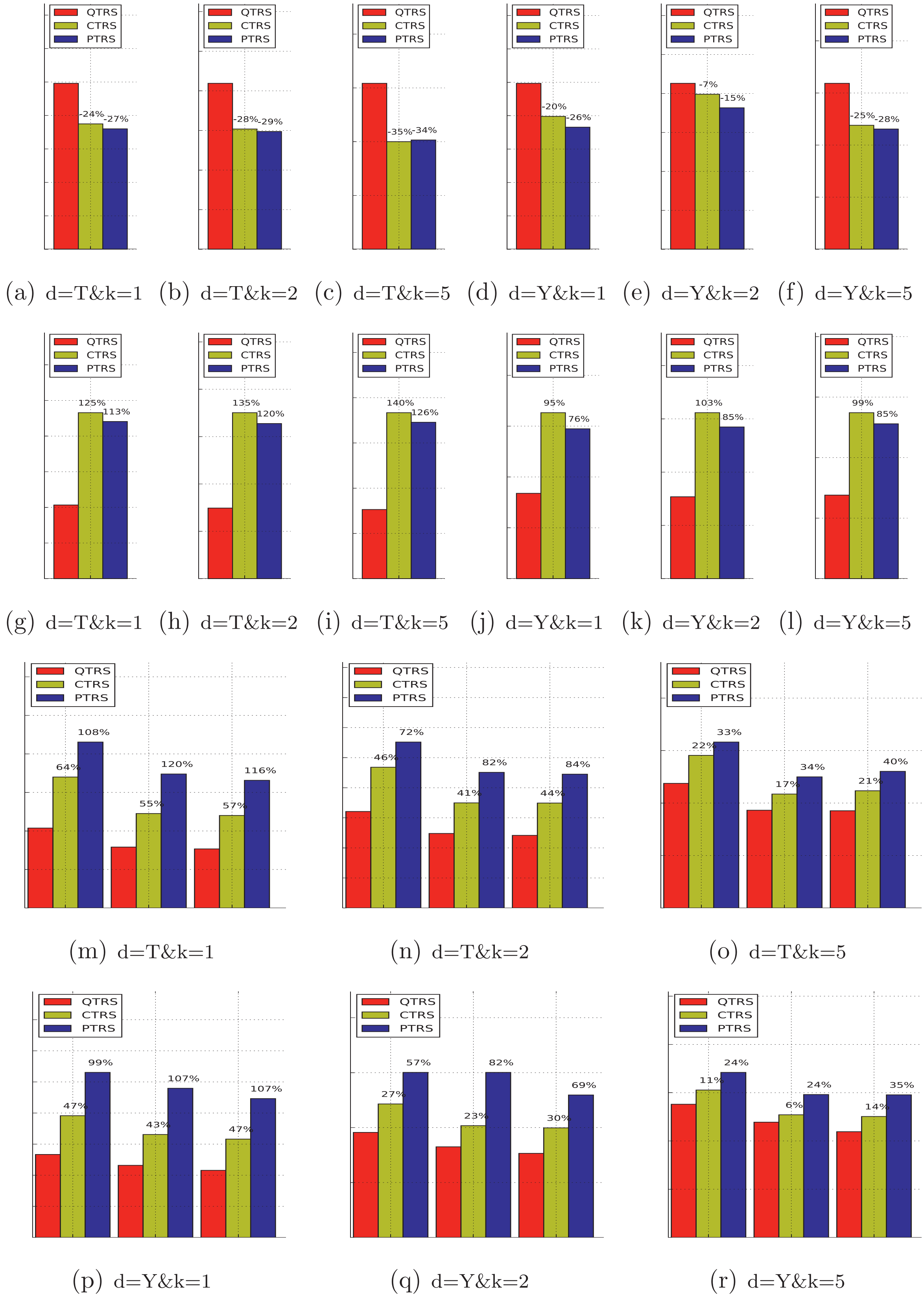
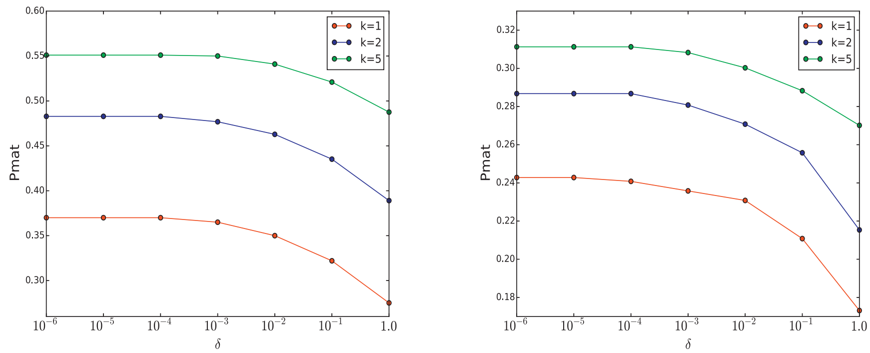


Fig. 2. Performance comparison of QTRS, CTRS and PTRS on Tripadvisor and Yelp datasets. “d=T(Y)” indicates on Tripadvisor (Yelp) dataset. “Pmat_c(p,s)” denotes the Pmat measurements with Cosine (Pearson, Spearman) correlation.



(a) On Tripadvisor dataset. (b) On Yelp dataset.

Fig. 3. Parameter sensitive analysis.

Table 3
Case Study (the aspects mentioned in the reviews are marked out with symbol [xxx]).

r_j^i	My daughter and I stayed here for two nights before a surgery she was having. We LOVED it! We live 100 miles from Houston and go often, but we had never stayed in this area of town. We will deffinatly stay again. The hotel is walking distance to great restraunts and entertainment *[location, food]*. EVERYONE of the hotel staff was happy and nice and smiling *[service]*. The valet even knew my name....it was awesome *[service]*. I will stay here again soon Oh and I loved the Einstine Bagels in the hotel also....it hit the spot!*[food]*
$\hat{\mathcal{R}}_{i,j}^{pers(0)}$	I have stayed at the Hyatt Downtown Houston many times during the last three years and have never been disappointed. The check-in staff is very friendly (although almost clinically friendly) and always helpful *[service]*. The location is great if you have business Downtown and there are a number of nice restaurants nearby (I like "Ibiza" a lot) *[location, food]*. The valet parking is easy and fast. I have eaten at the "Spindletop" restaurant in the hotel which was fine/good but nothing special *[food]*. I hear that the steak house in the hotel is supposed to be excellent (but have not tried it yet) *[food]*. All in all, the Hyatt is a great resting place for business travelers but offers nothing unique in terms of style, atmosphere or attention to the guest. That is just fine for me while on business. My expectations for holidays or special occasions are different but that's not relevant while assessing this hotel as a business traveler.
$\hat{\mathcal{R}}_{i,j}^{comp(0)}$	My recent (memorial day 2012) stay at the Hyatt Regency Houston was relaxing and enjoyable, and I 'd probably return – especially at the discount rate I enjoyed on this occasion *[price]*. But I noticed a few shortcomings which, while not at all critical, took some of the shine off the Hyatt brand, IMO. To wit: *my reservation for a smoking room had not been communicated to the front desk, resulting in an \$80 "upgrade" for two nights *[price, service]*. I was not going to argue about it, but it was an unnecessary aggravation*no refrigerator in the room *[room]*. This is a feature I think most travelers have come to expect in an upscale hotel room in the 21st century*the nicest business center I have ever seen – except for the \$.40/minute charge to access the Web *[business service, price]*. Another unexpected and inconvenient charge*a loud hum from the A/C system, combined with the atrium din I mention elsewhere, necessitated earplugs (which, on balance, were provided in the nightstand) I must stress, again, that my stay was comfortable overall – and that I feel I got a fair deal, based on my discount. I would not, however, pay rack rate: the extra charges and gaps in room features will have me looking more closely for a first-class room in downtown Houston next time around.
$\hat{\mathcal{R}}_{i,j}^{qual(0)}$	I stayed here for three nights. I asked for a quiet room and was given a "corner" room [sleep quality]. I am glad I did. The hotel has an issue with noise because the bar is in the center of the main floor and the rest of the hotel circles around it atrium style *[sleep quality]*. So all rooms have bar noise at night *[sleep quality]*. I had none. They were doing a sound survey while I was there...so they know about the problem and must be looking into solutions. My room was great *[room]*. Bed very comfortable and amenities were nice *[sleep quality]*. I was here for business and the room was booked and paid for so I have no idea about costs *[price]*.

- The top-1 review in $\hat{\mathcal{R}}_{i,j}^{pers}$ (denoted as $\hat{\mathcal{R}}_{i,j}^{pers(0)}$) is obtained by our method (i.e., PTRS). The review got 60 votes and covers three aspects: *location, service* and *food*. Although the number of votes and the number of covered aspects are both not very large, the review actually talks about the aspects u_i cares about.
- $\hat{\mathcal{R}}_{i,j}^{comp(0)}$ is obtained by CTRS method. It got 78 votes and covers more aspects: *price, room, location* and *business service*. Although the number of votes and the number of covered aspects are both a little higher than those of $\hat{\mathcal{R}}_{i,j}^{pers}$, the review fails to provide information about aspects *location* and *food*, which u_i cares about.
- $\hat{\mathcal{R}}_{i,j}^{qual(0)}$ is obtained by QTRS method. It got 134 votes and also covers three aspects: *price, sleep quality*, and *room*. Although the number of votes is much higher compared to the top-1 reviews in PTRS and CTRS, the review fails to provide information for aspects *location, service* and *food*, which u_i cares about.

Summing up, compared to other competitors, the top review set generated by our method provides more information about the product aspects that are important to the target user.

6. Discussion on PTRS

As the experimental results show, compared to other approaches, the top reviews obtained by our PTRS are much more correlated with product aspects important to the target users; thus, PTRS achieves personalized review selection. Still, PTRS suffers from the "cold start users" problem. PTRS needs past reviews of the target users to construct their aspect profiles for computing user similarities. Thus, PTRS could not select personalized reviews for the new users who have not submitted any reviews. In order to alleviate this problem, other sources of user data could be used for inferring the similarities between user aspect profiles. For example, it is worth studying whether two users who always buy the same products have similar aspect profiles. By this way, for a user without past reviews, we could use his/her purchase history to computer user similarities between him/her and other users. Then, PTRS can work for him/her. Otherwise, note that PTRS is computationally more expensive compared to CTRS. The extra computations are for calculating the aspect-importance distributions of users. Specifically, assume that the number of (user, product) queries is m and the average number of reviewers for each product is n, the time complexity of predicting aspect-importance distributions will be $O(mn)$.

7. Related work

The most related work to ours studies the prediction of personalized preferences in reviews {CITEpersBlhyprating1 persBlhyprating2. Approaches in this direction utilize users' "like" clicks on reviews and apply collaborative filtering to predict the users' preference on reviews. However, "like" clicks are typically very sparse on product commenting sites. Our work identifies the users' personal needs on the product-aspect level. Besides, our work also considers the coverage and quality characteristics of the selected reviews. Currently, owing to the lack of available datasets (the data in [19] are news comments and the dataset in [20] is not publicly available), we have not yet considered the users' "like" behaviors to retrieve personalized review sets.

There has been a substantial amount of work on automatically estimating the quality of a review for ranking purposes [4–8]. We have compared our approach to a ranking method, which disregards coverage and personalized importance of product aspects and found such a method inferior to both our method and the coverage-based ranking approach (CTRS) of Tsaparas et al. [1]. Our work inherits the merits of CTRS and also considers the quality of reviews in the selection process.

Related work in opinion summarization aims at extracting aspects (or features) of an product, and a short piece of text that summarizes the opinions on the different aspects [2,3,21–24]. For instance, Lappas and Gunopulos [2] consider the problem of finding a small set of reviews that cover all product attributes, in a similar spirit to [1]. Our approach is orthogonal to coverage-based selection and can in fact be used to also improve opinion summarization methods. For example, when selecting reviews to generate a summarization, the reviews providing more information about product aspects important to users can be given higher priority.

8. Conclusion and future work

8.1. Conclusion

In this paper, we considered the fact that different product aspects may have different importance to users; therefore, when a user wants to see the reviews on some product, it makes sense to prioritize reviews which talk about the product aspects that are the most important to the user. Our work is the first attempt on personalized top-review sets (TRS) selection for users. We select a TRS which includes not only reviews of high quality and coverage, but also are more focused on product aspects that are important to the user. To achieve this goal, we propose a method to automatically predict which product aspects are important to the target user (i.e., the *aspect-importance distribution*). Then, we utilize the predicted *aspect-importance distributions* to adjust the aspect weights during retrieving high-quality and high-coverage top reviews for the user. Our experiments show that our methodology selects reviews that focus more on the product aspects that are important to the user, without sacrificing coverage and high degree of quality.

8.2. Future work

Extending directions for our work presented in this paper include:

- We plan to seek opportunities of collaboration with real-world product-commenting platforms, in order to see the users' real response to the personalized review recommenders obtained by our method and competitors. We also intend to use our method for ranking reviews about music, movies, books and other kinds of items, in order to evaluate its generality.
- In this paper, we improve the previous work CTRS that considers both of quality and comprehensiveness of top reviews to also consider users' personalized preferences on product aspects. Compared to CTRS, some previous works paid attention to other characteristics such as diversification [25]. In our future work, we plan to improve these works to make the top reviews selected by adding users' personalized aspect preferences into their algorithms.
- We plan to use machine learning technologies [26,27] for improving the prediction of aspect-importance distributions.

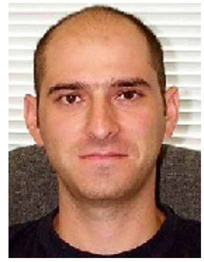
References

[1] P. Tsaparas, A. Ntoulas, E. Terzi, Selecting a comprehensive set of reviews., in: Proceedings of the 2011 ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), 2011.
[2] T. Lappas, D. Gunopulos, Efficient confident search in large review corpora., in: Proceedings of the 2010 European Conference on Machine Learning and Principles and Practice of Knowledge Discovery (ECML-PKDD), 2010.
[3] T. Lappas, M. Crovella, E. Terzi, Selecting a characteristic set of reviews., in: Proceedings of the 2012 ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), 2012.
[4] R. Zhang, T. Tran, Y. Mao, Opinion helpfulness prediction in the presence of words of few mouths., World Wide Web 15 (2) (2012) 117–138.
[5] A. Ghose, P. Ipeirotis, Estimating the helpfulness and economic impact of product reviews: mining text and reviewer characteristics, IEEE Trans. Knowl. Data Eng. 23 (2010) 10.
[6] Y. Liu, X. Huang, A. An, X. Yu, Modeling and predicting the helpfulness of online reviews., in: Proceedings of the 2008 IEEE International Conference on Data Mining (ICDM), 2008.
[7] O. Tsur, A. Rappoport, A fully unsupervised algorithm for selecting the most helpful book reviews., in: Proceedings of the 2009 International Conference on Weblogs and Social Media (ICWSM), 2009.
[8] C. Danescu-Niculescu-Mizil, G. Kossinets, J. Kleinberg, L. Lee, How opinions are received by online communities: a case study on Amazon.com helpfulness votes., in: Proceedings of the 2009 International Conference on World Wide Web (WWW), 2009.
[9] W. Yu, R. Zhang, X. He, C. Sha, Selecting a diversified set of reviews, in: Web Technologies and Applications, Springer, 2013, pp. 721–733.
[10] G. Chen, L. Chen, Augmenting service recommender systems by incorporating contextual opinions from user reviews, User Model. User-Adapt. Interact. 25 (3) (2015) 295–329.
[11] Z.-J. Zha, J. Yu, J. Tang, M. Wang, T.-S. Chua, Product aspect ranking and its applications, IEEE Trans. Knowl. Data Eng. 26 (5) (2014) 1211–1224.
[12] L. Zhang, B. Liu, Aspect and entity extraction for opinion mining, in: Data Mining and Knowledge Discovery for Big Data, Springer, 2014, pp. 1–40.
[13] J. Schafer, D.F.J. Herlocker, S. Sen, Collaborative filtering recommender systems, in: The Adaptive Web, ACM, 2007, pp. 291–324.
[14] W. Pan, A survey of transfer learning for collaborative recommendation with auxiliary data, Neurocomputing 177 (2016) 447–453.
[15] W. Tu, S. Sun, A subject transfer framework for eeg classification, Neurocomputing 82 (2012) 109–116.
[16] J. Li, M. Ott, B. Varadarajan, Identifying manipulated offerings on review portals., in: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2013.
[17] A. Tiroshi, S. Berkovsky, M. Kaafar, D. Vallet, T. Kuflik, Improving business rating predictions using graph based features., in: Proceedings of the 2014 International Conference on Intelligent User Interfaces (IUI), 2014.
[18] H. Wang, Y. Lu, C. Zhai, Latent aspect rating analysis on review text data: a rating regression approach., in: Proceedings of the 2010 ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), 2010.
[19] D. Agarwal, B. Chen, B. Pang, Personalized recommendation of user comments via factor models., in: Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2011.
[20] S. Moghaddam, M. Jamali, M. Ester, Review recommendation: personalized prediction of the quality of online reviews., in: Proceedings of the 2011 ACM International Conference on Information and Knowledge Management (CIKM), 2011.
[21] M. Hu, B. Liu, Mining opinion features in customer reviews., in: Proceedings of the 2004 AAAI Conference on Artificial Intelligence (AAAI), 2004.
[22] Y. Lu, C. Zhai, N. Sundaresan, Rated aspect summarization of short comments., in: Proceedings of the 2009 International Conference on World Wide Web (WWW), 2009.
[23] A. Popescu, B. Nguyen, O. Etzioni, Extracting product features and opinions from reviews., in: Proceedings of the 2005 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2005.
[24] M. Zimmermann, E. Ntoutsi, M. Spiliopoulou, Discovering and monitoring product features and the opinions on them with opinstream, Neurocomputing 150 (2015) 318–330.
[25] R. Krestel, N. Dokoohaki, Diversifying customer review rankings, Neural Netw. 66 (2015) 36–45.
[26] R.S. Michalski, J.G. Carbonell, T.M. Mitchell, Machine Learning: An Artificial Intelligence Approach, Springer Science & Business Media, 2013.

[27] O. Arqub, Adaptation of reproducing kernel algorithm for solving fuzzy Fredholm–Volterra integro differential equations, Neural Comput. Appl. (2015) 1–20.

Wenting Tu is currently an assistant professor at the department of computer science, Shanghai University of Finance and Economics. She received her PhD degree in computer science from University of Hong Kong (HKU) in 2016. Before that, She received the M. S. degree from the Department of Computer Science and Technology, East China Normal University in 2012. Her research interests include data mining, machine learning and nature language processing.

David Wai-lok Cheung received the M.Sc. and Ph.D. degrees in computer science from Simon Fraser University, Canada, in 1985 and 1989, respectively. Since 1994, he has been a faculty member of the Department of Computer Science in The University of Hong Kong. His research interests include database, data mining, database security and privacy. He was the Program Committee Chairman of PAKDD 2001, Program Co-Chair of PAKDD 2005, Conference Chair of PAKDD 2007 and 2011, Conference Co-Chair of CIKM 2009 and Conference Co-Chair of PAKDD 2011, Program Co-Chair of ER 2012, and Conference Co-Chair of PAKDD 2015.



Nikos Mamoulis received a diploma in Computer Engineering and Informatics in 1995 from the University of Patras, Greece, and a Ph.D. in Computer Science in 2000 from the Hong Kong University of Science and Technology. He is currently a professor at the Department of Computer Science, University of Hong Kong, which he joined in 2001. His research focuses on management and mining of complex data types, privacy and security in databases, and uncertain data management. He served as PC member in more than 80 international conferences on data management and mining. He served as an associate editor for IEEE TKDE and the VLDB Journal.