Recommendation for New Users with Partial Preferences by Integrating Product Reviews with Static Specifications

Feng Wang, Weike Pan, and Li Chen

Department of Computer Science, Hong Kong Baptist University
Hong Kong, China
{fwang,wkpan,lichen}@comp.hkbu.edu.hk

Abstract. Recommending products to new buyers is an important problem for online shopping services, since there are always new buyers joining a deployed system. In some recommender systems, a new buyer will be asked to indicate her/his preferences on some attributes of the product (like camera) in order to address the so called cold-start problem. Such collected preferences are usually not complete due to the user's cognitive limitation and/or unfamiliarity with the product domain, which are called *partial preferences*. The fundamental challenge of recommendation is thus that it may be difficult to accurately and reliably find some like-minded users via collaborative filtering techniques or match inherently preferred products with content-based methods. In this paper, we propose to leverage some auxiliary data of online reviewers' aspect-level opinions, so as to predict the buyer's missing preferences. The resulted user preferences are likely to be more accurate and complete. Experiment on a real user-study data and a crawled Amazon review data shows that our solution achieves better recommendation performance than several baseline methods.

Keywords: New users, partial preferences, product recommendation, consumer reviews, aspect-level opinion mining, static specifications.

1 Introduction

The importance of recommendation as an embedded component in various online shopping services has been well recognized [2]. Most recommendation algorithms are designed to make use of explicit or implicit feedbacks of experienced users. However, new buyers join a typical online service everyday, who usually have no explicit ratings and/or little implicit behaviors. Facing such a new-user recommendation problem, some deployed systems ask the buyer to indicate some preferences on certain attributes of the product [3,7], such as the camera's brand, price, resolution, etc. However, the limitation of such works is that the efforts required from the buyer would be inevitably high. Moreover, most buyers are in reality not able to state their full preferences (say over all attributes) due to their cognitive limitation and/or unfamiliarity with the product domain, even

S. Carberry et al. (Eds.): UMAP 2013, LNCS 7899, pp. 281–288, 2013.

when they are involved in a conversational interaction with the system [6,18]. The challenging issue is then how to predict the buyer's missing preferences on un-stated attributes, which is actually for solving the *partial preferences* problem [10]. The weakness of classical model-based and memory-based algorithms in collaborative filtering is that they can not build collaborative relationships among users without users' feedbacks [11,16]. The content-based methods may also fail to accurately find matching products when users' preferences are only given on a subset of attributes [13,15]. In the traditional artificial intelligent systems, some logic-oriented approaches were proposed for representing and reasoning about user preferences [10,14]. For instance, [10] presents a hybrid of quantitative and qualitative approach grounded on multi-attribute utility theory (MAUT) to identify sup-optimal alternatives. However, the approach's practical performance in the online environment is limited due to its high time complexity.

Therefore, in this paper, we propose a novel preference enrichment framework, which aims to complete a new buyer's preferences by incorporating product reviewers' aspect-level opinions and attributes' static specifications. Specifically, by integrating with the fine-grained opinion mining results of textual reviews, we target to find like-minded reviewers for a target new buyer and hence enrich the buyer's preferences on all attributes. Indeed, the advantages of reviews are that: 1) product reviews are broadly accessible over the internet. Therefore, even for a new system, it can extract product reviews from similar sites (like from Amazon) to serve its buyers; 2) reviews to a product can truly reflect the reviewer's preferences on various aspects of the product, as they are based on her/his post-usage evaluation experiences. Thus, it is expected that the incorporation of product reviews can bring *true* user preferences so as to ideally augment the system's recommendation accuracy for the current new buyer. To the best of our knowledge, though recently there are increasing attentions placed to exploit the values of product reviews in recommender systems [12,21,22], the aspect-level opinions have been rarely investigated for addressing the *partial preferences* problem. In our previous work [20], we emphasized mining reviewers' similarity network by revealing their weights placed on different aspects, but did not map their opinions to the attributes' static specifications for identifying their value preferences. Therefore, in this paper, our main interest is in exploiting such information to particularly predict new buyers' missing preferences.

Our contributions can be summarized as follows: 1) we envision product reviews as valuable resource of other users' preference information to enrich the current buyer's preferences; 2) we study how to leverage reviewers' aspect-level opinions, by mapping them to the attributes' static specifications, for aiding the product recommendation; 3) we conduct an empirical test of the proposed approach on a real user-study data and a crawled Amazon review data, which shows the outperforming accuracy of our solution against several baseline methods in the real-world setting (i.e., digital camera recommendation).

Table 1. Some notations used in the paper

	Notation	Description
Product	$p = 1, 2, \ldots, m$	product, e.g., Casio EX-Z55 DC
	$a = 1, 2, \ldots, k$	product attribute, e.g., *weight* = 129.9
	$\mathbf{x}_p = [x_{pa}]_{k\times 1} \in \mathbb{R}^{k\times 1}$	product profile
Buyer	$u = 1, 2, \ldots, n$	user
	$\boldsymbol{\phi}_u = [\phi_{ua}]_{k\times 1}$	user's preference, e.g., *weight* < 200
	$\mathbf{y}_u = [y_{ua}]_{k\times 1} \in \{0, 1\}^{k\times 1}$	user preference indicator ($y_{ua} = 1$ if user u states preference on attribute a)
	$choice(u) \in \{1, 2, \ldots, m\}$	user's target choice (ground truth)
Reviewer	$\tilde{u} = 1, 2, \ldots, \tilde{n}$	reviewer
	$\tilde{\boldsymbol{\phi}}_{\tilde{u}} = [\tilde{\phi}_{\tilde{u}a}]_{k\times 1}$	reviewer's preferences on various attributes
	$\tilde{r}_{\tilde{u}p} \in \{1, 2, 3, 4, 5\}$	reviewer's rating to a product

2 Problem Definition and Methodology

We have n new buyers and m products, where each buyer indicates preferences on a subset of product attributes, e.g., *weight* $< 200g$, *price* $<$ \$300. We also have some auxiliary data of online reviews on those m products. As mentioned before, our goal is to enrich the new buyer's preferences and then recommend a personalized ranking list of products to him/her. We list some notations used in the paper in Table 1.

Our proposed solution, called *preference completion and ranking* (henceforth called CompleteRank), mainly contains the following three steps.

Step 1: Aspect-Level Opinion Mining. The online product reviews written by the users who previously purchased products usually contain some positive and/or negative opinions on certain aspects of a product. Thus, it is straightforward to assume that these aspect-level opinions can reflect the inherent preferences of the author (i.e., the reviewer) on the product's attributes (note that *attributes* refer to the product's static specifications, while *aspects* are features discovered from reviews. The latter is mapped to the former through a predefined dictionary). Inspired from this observation, we emphasize the usage of aspect-level opinion mining outcomes for predicting a new user's missing preferences. Since reviews are written in natural language, we need to first extract the aspects and opinions from a large amount of reviews automatically. This issue was addressed in our prior work that is capable of identifying the aspect-level opinions from a review [19,20]. Basically, there are three sub-steps: (1) identify all (aspect, opinion) pairs in a review through the Part-of-Speech tagger[1] (which is for extracting frequent nouns and noun phrases as aspect candidates), syntactic dependency parser[2] (which is for identifying opinion words)

[1] http://nlp.stanford.edu/software/tagger.shtml

[2] http://nlp.stanford.edu/software/lex-parser.shtml

and WordNet [9] (for grouping synonymous aspects). (2) Quantify the opinion's sentiment strength (also called *polarity*) by applying SentiWordNet [8]. Formally, the aspect-level opinion is classified as negative (-1) or positive (1). (3) Map the opinion to the attribute's static specification in a structured form *(attribute, opinion, specification)*. For example, *("weight", 1, 200g)* indicates that the reviewer expresses *positive* opinion on the product's weight that is 200g, which can further imply that this reviewer's value preference on the attribute "weight" lies in a range that contains 200g.

Step 2: Preference Completion. For each new buyer u, we complete her/his preferences with the help of some like-minded reviewers' preferences,

$$\overrightarrow{\tilde{\phi}_{ua}} = \begin{cases} (\overrightarrow{\phi_{ua}} + \sum_{\tilde{u}\in N_u} \bar{s}_{u\tilde{u}} \overrightarrow{\tilde{\phi}_{\tilde{u}a}})/2, & \text{if } \phi_{ua} \text{ is not missing} \\ \sum_{\tilde{u}\in N_u} \bar{s}_{u\tilde{u}} \overrightarrow{\tilde{\phi}_{\tilde{u}a}}, & \text{otherwise} \end{cases}, \qquad (1)$$

where $\bar{s}_{u\tilde{u}} = \frac{s_{u\tilde{u}}}{\sum_{\tilde{u}\in N_u} s_{u\tilde{u}}}$ is the normalized similarity between buyer u and reviewer $\tilde{u}$. The similarity is calculated as $s_{u\tilde{u}} = \sum_{a=1}^{k} y_{ua} \times \cos(\overrightarrow{\phi_{ua}}, \overrightarrow{\tilde{\phi}_{\tilde{u}a}})$, where $\overrightarrow{\phi_{ua}}$ and $\overrightarrow{\tilde{\phi}_{\tilde{u}a}}$ are respectively the vector representations of the buyer u and reviewer $\tilde{u}$'s value preferences on the attribute a. For instance, suppose a is the camera's weight which is classified into 8 intervals: $[0, 200)$, $[200, 400)$, ..., and $[1200, 1400)$. If a reviewer's preference on "weight" is in the range $[200, 400)$, her/his corresponding vector representation $\overrightarrow{\tilde{\phi}_{\tilde{u}a}}$ is $[0, 1, 0, 0, 0, 0, 0, 0]$.

Thus, if the buyer's preference on an attribute a is not missing, similar reviewers' preferences regarding this attribute are used to adjust the buyer's preference on it, so as to fuse the reviewers' collective preferences. Otherwise, they are adopted to predict the buyer's preference on that attribute, i.e., which interval(s) her/his value preference lies in. We illustrate the preference completion procedure in Figure 1. Note that we use $|N_u| = 300$ for the size of group of like-minded reviewers in our experiment.

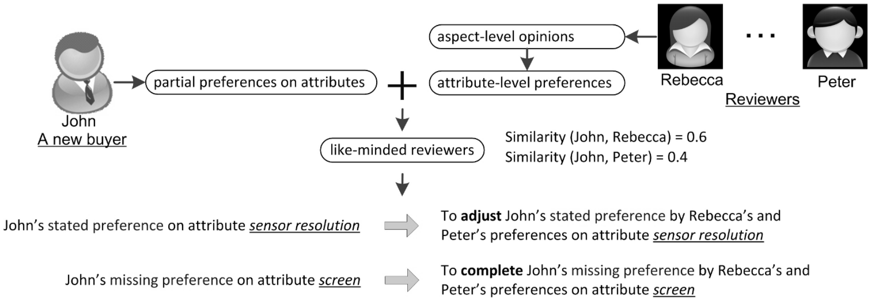
aspect-level opinions
Rebecca
...
Peter
Reviewers
John
A new buyer
partial preferences on attributes
+
attribute-level preferences
like-minded reviewers
Similarity (John, Rebecca) = 0.6
Similarity (John, Peter) = 0.4
John's stated preference on attribute sensor resolution
To adjust John's stated preference by Rebecca's and Peter's preferences on attribute sensor resolution
John's missing preference on attribute screen
To complete John's missing preference by Rebecca's and Peter's preferences on attribute screen

Fig. 1. Illustration of preference completion procedure with an example

Step 3: Ranking and Recommendation. With the enriched user preferences, we can then calculate the matching score between a buyer u and a product p,

$$M_{up} = \frac{1}{k} \sum_{a=1}^{k} match_w(\bar{\phi}_{ua}, x_{pa}) \tag{2}$$

where $match_w(\bar{\phi}_{ua}, x_{pa}) = \langle \overrightarrow{\bar{\phi}_{ua}}, \overrightarrow{x_{pa}} \rangle$ is the inner product of the expanded vectors w.r.t. attribute a. The obtained matching scores can then be used to rank products. The ones with highest scores are recommended to the target buyer.

3 Experimental Results

3.1 Data and Evaluation Metric

We have two data sets, one collected from a previous user study [4] and the other from Amazon review data. In our user study data, there are 57 users ($n = 57$), and 64 digital cameras ($m = 64$) where each product has 8 attributes ($k = 8$). Each user explicitly indicated her/his preferences on the product's attributes. Each user was also asked to check all products and carefully chose one product as her/his favorite product, denoted as $choice(u)$ (i.e., the user's target choice). For each product, we crawled the corresponding reviews from the Amazon website (http://www.amazon.com/). The total number of reviews is 4904 as from 4904 reviewers (since each reviewer posted only one review among those products). In our experiment, for each of these 57 users, we randomly select $2, 4$, or 6 of her/his attribute preferences to represent the simulated buyer's *partial* preferences (e.g., 2 means that the buyer just stated preferences on 2 attributes).

For each user u, there is a target choice in the product set, i.e., $choice(u)$, which is taken as the ground truth in our evaluation. We use hit ratio of the recommended top-N products to evaluate the recommendation accuracy, $H@N = \frac{1}{n} \sum_{u=1}^{n} \delta(position(choice(u)) \leq N)$, where $choice(u)$ is the target choice of user u, $position(choice(u))$ denotes its ranking position, and n is the number of users. Note that $\delta(z) = 1$ if z is true and $\delta(z) = 0$ otherwise. In our experiment, we use $N = 10$, since a typical user only checks a few products which are placed in top positions [5].

3.2 Baselines

We compare our proposed solution with the following four baseline methods (most of which are from related literatures).

Random. We randomly rank the products for each target user. The result is calculated as $N/m = 10/64 = 0.1563$, denoting the probability that the user's target choice is ranked among top 10.

PopRank. We calculate the popularity of each product among the reviewers. A product is usually considered as *preferred* by a reviewer if the rating is larger

Table 2. The recommendation accuracy (hit ratio) of CompleteRank and other baselines. Note that for PartialRank, HybridRank and CompleteRank, we randomly took 2, 4, 6 attributes (each under five runs) to simulate *partial preferences.*

Method	Given 2	Give 4	Given 6	Given 8
Random	0.1563	0.1563	0.1563	0.1563
PopRank	0.2456	0.2456	0.2456	0.2456
PartialRank	0.1825$_{\pm 0.0457}$	0.2211$_{\pm 0.0342}$	0.2772$_{\pm 0.0288}$	0.3158
HybridRank	0.2386$_{\pm 0.0440}$	0.2456$_{\pm 0.0447}$	0.2947$_{\pm 0.0192}$	0.2982
CompleteRank	**0.2807**$_{\pm 0.0372}$	**0.3088**$_{\pm 0.0457}$	**0.3158**$_{\pm 0.0277}$	**0.3333**

than 3 in 5-star numerical ratings [17]. The popularity of the product p among the reviewers can then be estimated as, $P_p = \frac{1}{\tilde{n}} \sum_{\tilde{u}=1}^{\tilde{n}} \delta(\tilde{r}_{\tilde{u}p} > 3)$. The obtained popularity scores $0 \leq P_p \leq 1$ are used to rank all products. Note that PopRank is not a personalized method since the popularity is user independent.

PartialRank. For each user u and product p, we calculate the matching score between the user's stated (partial) preferences and the product's profile, $M_{up} = \frac{1}{k} \sum_{a=1}^{k} y_{ua} \times match(\phi_{ua}, x_{pa})$, where $match(\phi_{ua}, x_{pa}) = 1$ if the attribute's static specification x_{pa} satisfies the user preference ϕ_{ua}, and $match(\phi_{ua}, x_{pa}) = 0$ otherwise. The obtained matching scores, $0 \leq M_{up} \leq 1$ with $p = 1, \ldots, m$, can then be used to rank the products for user u.

HybridRank. For each attribute a of product p, we can calculate the average opinion score from the reviewers, i.e. $opinion(p, a) \in [-1, 1]$, and the product p's overall opinion score via the method proposed in [1], $O_{up} = \frac{1}{k} \sum_{a=1}^{k} y_{ua} \times opinion(p, a)$. Then, with the preference matching score M_{up} (from PartialRank) and opinion score O_{up}, a hybrid score is produced for the product p, $H_{up} = \frac{1}{2}(M_{up} + O_{up})$. The obtained scores, $-1 \leq H_{up} \leq 1$ with $p = 1, \ldots, m$, are used to rank the products for user u.

3.3 Summary of Experimental Results

The results are shown in Table 2, from which we can have the following observations, (1) our proposed solution CompleteRank is much better than all baselines, which clearly shows the effectiveness of our preference enrichment idea, especially for the buyers with partial preferences; (2) PopRank is better than Random, which demonstrates the usefulness of incorporating online review data for augmenting new-user recommendation; (3) PartialRank performs worse than PopRank given 2 and 4 attribute preferences, but better than PopRank when given 6 and 8 attribute preferences, which shows the effect of taking into account the current user's preferences (especially when they are nearly complete) on increasing recommendation accuracy; and (4) HybridRank performs better than PartialRank in most cases, which shows the usefulness of combining the product's static specifications (by matching to users' preferences) and reviewers' opinions, though it is still worse than our solution.

4 Conclusions and Future Work

In this paper, we propose a preference enrichment approach, CompleteRank, via incorporating the mined reviewers' aspect-level opinions on products' static specifications. The completed preferences of a new user are then used to match the products' profiles, by which the products with highest matching scores are recommended to the target user. Experimental results show that our solution can provide more accurate personalized recommendation than several baseline methods. For future work, we plan to further integrate reviewers' weights (i.e., the importance degrees) placed on attributes (as learnt from our previous work [20]), so that a weighted value preference model might be built for each reviewer. The preference enrichment framework for new buyers could hence be additionally improved by leveraging these heterogeneous types of review data.

Acknowledgements. This research work was supported by Hong Kong Research Grants Council under project ECS/HKBU211912.

References

1. Aciar, S., Zhang, D., Simoff, S., Debenham, J.: Informed recommender: Basing recommendations on consumer product reviews. IEEE Intelligent Systems 22(3), 39–47 (2007)
2. Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. IEEE Trans. on Knowl. and Data Eng. 17(6), 734–749 (2005)
3. Butler, J.C., Dyer, J.S., Jia, J., Tomak, K.: Enabling e-transactions with multi-attribute preference models. European Journal of Operational Research 186(2), 748–765 (2008)
4. Chen, L., Pu, P.: A cross-cultural user evaluation of product recommender interfaces. In: Proceedings of the 2008 ACM Conference on Recommender Systems, RecSys 2008, pp. 75–82. ACM, New York (2008)
5. Chen, L., Pu, P.: Users' eye gaze pattern in organization-based recommender interfaces. In: Proceedings of the 16th International Conference on Intelligent user Interfaces, IUI 2011, pp. 311–314. ACM, New York (2011)
6. Chen, L., Pu, P.: Critiquing-based recommenders: survey and emerging trends. User Modeling and User-Adapted Interaction 22(1-2), 125–150 (2012)
7. Edwards, W.: Social utilities. Engineering Economist 6, 119–129 (1971)
8. Esuli, A., Sebastiani, F.: Sentiwordnet: A publicly available lexical resource for opinion mining. In: Proceedings of the 5th Conference on Language Resources and Evaluation, LREC 2006, pp. 417–422 (2006)
9. Fellbaum, C.: WordNet: An Electronic Lexical Database. MIT Press, Cambridge (1998)
10. Ha, V., Haddawy, P.: A hybrid approach to reasoning with partially elicited preference models. In: Proceedings of the 15th Conference on Uncertainty in Artificial Intelligence, UAI 1999, pp. 263–270. Morgan Kaufmann Publishers Inc., San Francisco (1999)

11. Koren, Y.: Factorization meets the neighborhood: a multifaceted collaborative filtering model. In: Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2008, pp. 426–434. ACM, New York (2008)
12. Levi, A., Mokryn, O., Diot, C., Taft, N.: Finding a needle in a haystack of reviews: cold start context-based hotel recommender system. In: Proceedings of the 6th ACM Conference on Recommender Systems, RecSys 2012, New York, NY, USA, pp. 115–122 (2012)
13. Liu, Q., Chen, T., Cai, J., Yu, D.: Enlister: baidu's recommender system for the biggest chinese q/a website. In: Proceedings of the Sixth ACM Conference on Recommender Systems, RecSys 2012, pp. 285–288. ACM, New York (2012)
14. Nguyen, T.A., Do, M., Gerevini, A.E., Serina, I., Srivastava, B., Kambhampati, S.: Generating diverse plans to handle unknown and partially known user preferences. Artif. Intell. 190, 1–31 (2012)
15. Pazzani, M.J., Billsus, D.: Content-based recommendation systems. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) Adaptive Web 2007. LNCS, vol. 4321, pp. 325–341. Springer, Heidelberg (2007)
16. Rendle, S.: Factorization machines with libfm. ACM Trans. Intell. Syst. Technol. 3(3), 57:1–57:22 (2012)
17. Sindhwani, V., Bucak, S.S., Hu, J., Mojsilovic, A.: A family of non-negative matrix factorizations for one-class collaborative filtering. In: The 1st International Workshop on Recommendation-based Industrial Applications held in the 3rd ACM Conference on Recommender Systems, RecSys: RIA 2009 (2009)
18. Viappiani, P., Faltings, B., Pu, P.: Preference-based search using example-critiquing with suggestions. J. Artif. Int. Res. 27(1), 465–503 (2006)
19. Wang, F., Chen, L.: Recommendation based on mining product reviews' preference similarity network. In: The 6th Workshop on Social Network Mining and Analysis, 2012 ACM SIGKDD Conference on Knowledge Discovery and Data Mining, SNA-KDD 2012 (2012)
20. Wang, F., Chen, L.: Recommending inexperienced products via learning from consumer reviews. In: Proceedings of the 2012 IEEE/WIC/ACM International Conferences on Web Intelligence, WI 2012, pp. 596–603. IEEE Computer Society, Washington, DC (2012)
21. Yates, A., Joseph, J., Popescu, A.-M., Cohn, A.D., Sillick, N.: Shopsmart: product recommendations through technical specifications and user reviews. In: Proceedings of the 17th ACM Conference on Information and Knowledge Management, CIKM 2008, pp. 1501–1502. ACM, New York (2008)
22. Zhang, W., Ding, G., Chen, L., Li, C., Zhang, C.: Generating virtual ratings from chinese reviews to augment online recommendations. ACM Trans. Intell. Syst. Technol. 4(1) (2013)