Collaborative Compound Critiquing

Haoran Xie, Li Chen, and Feng Wang

Department of Computer Science, Hong Kong Baptist University
hrxie2@gmail.com, {lichen,fwang}@comp.hkbu.edu.hk

Abstract. Critiquing-based recommender systems offer users a conversational paradigm to provide their feedback, named *critiques*, during the process of viewing the current recommendation. In this way, the system is able to learn and adapt to the users' preferences more precisely so that better recommendation could be returned in the subsequent iteration. Moreover, recent works on experience-based critiquing have suggested the power of improving the recommendation efficiency by making use of relevant sessions from other users' histories so as to save the active user's interaction effort. In this paper, we present a novel approach to processing the history data and apply it to the compound critiquing system. Specifically, we develop a history-aware collaborative compound critiquing method based on preference-based compound critique generation and graph-based similar session identification. Through experiments on two data sets, we validate the outperforming efficiency of our proposed method in comparison to the other experience-based methods. In addition, we verify that incorporating user histories into compound critiquing system can be significantly more effective than the corresponding unit critiquing system.

Keywords: Conversational recommender systems, history-aware compound critiquing.

1 Introduction

Product recommender systems (RS) have become critical part of many online e-commerce systems as they can assist users in effectively navigating through the large product space for making accurate choices. Specifically, critiquing-based recommender systems offer users a conversational paradigm to provide their feedback to the current recommendation, named *critiques* (e.g., "slower CPU" or "cheaper price" to a laptop), so as for the system to be able to refine its understanding of users' needs and return better recommendation to them in the next cycle [6]. Particularly, it has been found that such kind of system is highly competent to support users in revising and completing preferences in the high-risk product domains (such as cars, laptops, houses) given that users are often unable to fully state their preferences at the start due to the unfamiliarity with the products [1]. Prior works showed that a certain amount of conversational cycles is often required till the user locates her/his target choice [15,16]. The most critical question is then how to minimize users' interaction effort, without

V. Dimitrova et al. (Eds.): UMAP 2014, LNCS 8538, pp. 254–265, 2014.

compromising the decision accuracy that they can obtain by using the critiquing-based RS.

Recently, some researchers have attempted to utilize other users' critiquing histories to serve the current user. For example, in [13], the relevant historical sessions between other users and the current user are identified according to the number of their overlapping critiques, and then the accepted items in similar sessions will be considered for recommending to the current user. Later, this *experience-based* approach has been improved by incorporating the compatibility score [13] and item similarity [17]. However, there are two main limitations of these related works: 1) they neglect the sequence of items/critiques in identifying similar sessions; 2) they are applied to unit critiquing system only. Indeed, from the aspect of critiquing unit, there are two major types of critiquing-based RS: *unit critiquing* and *compound critiquing* [6]. In the former system (e.g., FindMe [3]), users are allowed to critique a single attribute at a time, like "faster processor" or "cheaper" to an example laptop, while in the compound critiquing system, each critique can be a combination of multiple unit critiques which operates over multiple attributes simultaneously (e.g., "different manufacture, lower processor speed and cheaper") [15]. The experiment done in [16,12] showed that the total number of recommendation cycles can significantly decrease when users selected the compound critiques. It is hence meaningful to study how to incorporate other users' critiquing histories into the standard compound critiquing system, so as to further save the current user's interaction effort.

Therefore, in this paper, we present a novel approach, named *collaborative compound critiquing*, to achieve the above-mentioned goal. From the perspective of method improvement, rather than simply counting the overlapping critiques among users' sessions, we develop a graph-based similarity measure to identify similar sessions based on other users' critiquing history data. Moreover, a new product ranking function is proposed by taking sub-session similarity into consideration. In the experiments, we compared our method with related approaches in both compound and unit critiquing systems. The results show that our method can significantly outperform the existing approaches.

The remainder of this paper is structured as follows. In Section 2, we review the related researches about critiquing-based recommender systems. Our proposed methodology is introduced in Section 3, which is divided into subsections including *compound critique generation*, *similar session identification*, and *item recommendation*. The experiment setup and results analysis are given in Section 4. We finally summarize this research and indicate its future research directions in Section 5.

2 Related Work

In this section, we review the related work from two aspects: critiquing unit and critiquing history-awareness. Note that the mentioned works all aim to suggest a set of critiques to the currently recommended item during each interaction cycle for users to select. The user-selected critique is then taken as feedback for the system to recommend a new item in the next cycle.

As mentioned before, there are two major kinds of **critiquing unit** in current critiquing-based recommender systems: *unit critiquing* and *compound critiquing*. The unit critiquing refers to a simple quantity or quality based feedback on a single attribute. According to [10], some users are willing to make unit critiques due to the simplicity and low cognitive load that they consume. FindMe system was the first well-known unit critiquing system [3]. It uses knowledge about the product domain to help users navigate through the multi-dimensional space, by proposing several pre-designed unit critiques (e.g., “cheaper”, “bigger”, and “nicer”) which are called “tweaks” in their system, for users to select [3]. When a user finds the current recommendation short of her/his expectation and responds to a tweak, the remaining candidates will be filtered to leave only those candidates satisfying this tweak. In another related system ATA (Automated Travel Assistant), two *extrema*, i.e., the cheapest trip and the best non-stop trip, are suggested to the user [8].

However, considering that the unit critiques might mislead users that individual features are independent and hence make them be engaged in unnecessary cycles when searching for their desired product [4], Dynamic Critiquing proposed to generate a set of compound critiques, each of which operates over multiple attributes simultaneously (e.g., “different Manufacture, lower processor speed and cheaper”) [15,11]. With such compound critique, users can see which attributes are highly dependent between each other. The compound critiques are concretely computed by discovering the recurring sets of unit differences between the currently recommended item and the remaining products through association rule mining [15]. Zhang and Pu [18] further improved this approach by adapting the generation and selection of compound critiques to users’ preferences which are modeled based on Multi-Attribute Utility Theory (MAUT) [7]. In comparison, Preference-based Organization technique [5] can be considered as a combination of the advantages of Dynamic Critiquing [15] and MAUT-based compound critiques [18]. It can not only dynamically generate critiques adaptive to users’ MAUT-based preference model, but also apply the association rule mining tool to discover compound critiques being representative of the remaining dataset. In addition, the critiques and their contained products are diversified so as to assist users in refining and accumulating their preferences more effectively.

From the aspect of **critiquing history-awareness**, some researchers have recently attempted to reuse past users’ critiquing histories to serve the current user, so as to save her/his interaction effort. For example, in [13], considering that the critiquing histories might carry valuable information about other users’ attribute preferences, they proposed the *experience-based critiquing* to harness these histories to guide the critiquing process for the current user. [9] further improved this work when selecting items as recommendation candidates, which include not only ones finally accepted by like-minded users who have critiquing sessions relevant to the current user, but also the items recommended during these sessions. More lately, [17] incorporated the item similarity between two sessions into discovering similar sessions, which gained better performance than the other approaches in terms of cycle reduction. However, these approaches mainly focus on improving unit critiquing system. Their methods are also limited in taking into account the sequential relationship between items/critiques in one session when identifying similar sessions.

3 Methodology

Research Problem Formulation. We focus on the research problem of how to realize *collaborative compound critiquing*. Formally, it can be modeled as a mapping function θ:

$$\theta : I \times Q \times S \rightarrow R \tag{1}$$

where I is the set of all items in the system, Q is the current critiquing session, S is the set of critiquing sessions from other users, and R is the set of ranking scores over all items. In each interaction cycle, the system will recommend the item with the highest ranking score. The process continues until the user accepts one item as the final choice. Specifically, the procedure of computing recommendation contains three sub-processes (see Figure 1): *compound critique generation*, *similar session identification*, and *item recommendation*. In the following, we will in detail describe how each sub-process is conducted in our system.

Compound Critique Generation. Because the Preference-based Organization technique [5] was demonstrated achieving the highest critique prediction accuracy and recommendation accuracy relative to the other compound critique generation approaches, in this work, we aim to enhance this compound critiquing system by incorporating other users' critiquing histories. The definition of compound critique in such system is as follows.

Definition 1. *(Compound Critique). The compound critique, denoted as C_i, is an element in the power set of all elemental (unit) critiques, i.e., $C_i \in \mathcal{P}(\mathbb{C})$, where $c_n \in \mathbb{C}$ is a triplet in the form of attribute, operator, and value:*

$$c_n = (attribute_n, operator_n, value_n)$$

where $attribute_n$ is the attribute (category) for critiquing, $operator_n$ is an element in the operator set $\{=, \neq, >, <\}$[1], and $value_n$ is the value for the operator. Note that each element c_n in a compound critique C_i is a unit critique. An example of compound critique for the laptop is "CPU speed $> 2.30GHz$; price $< HK\$6000$", which is formed of two unit critiques.

The preference-based organization technique [5] is composed of two main steps. Firstly, a set of frequently occurring attribute sets ($attribute_n$, $operator_n$, $value_n$) (where the *value* is of the current recommendation) among items in the remaining dataset are discovered by the association rule mining Aprior algorithm, which are then taken as the critique candidates. Next, each candidate is computed with a score via the function ϕ:

$$\phi(C_i) = U(C_i) \times D(C_i, SC) \tag{2}$$

[1] $=, >, <$ are used for numerical attributes, and $=, \neq$ are for categorical ones.

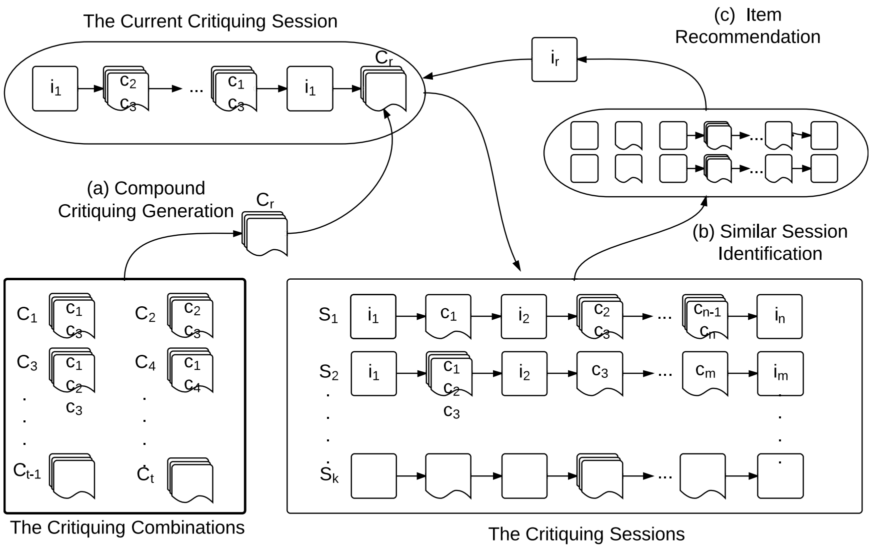

Fig. 1. The framework of our proposed *collaborative compound critiquing* system.

which takes into account both tradeoff utility $U(C_i)$ of the currently considered critique candidate C_i and its diversity with the critiques selected so far $D(C_i, SC)$. Specifically, the utility function U is defined as:

$$U(C_i) = \sum_{n=1}^{|C_i|} (\alpha_n * w(attribute_n)) \times \frac{1}{SR(C_i)} \sum_{i \in SR(C_i)}^{|SR(C_i)|} u(i) \tag{3}$$

where each unit critique in C_i is associated with a trade-off parameter α_n set as default value 0.75 if better than the current recommendation's attribute value, or 0.25 if worse, and w is the attribute's relative importance. $\frac{1}{SR(C_i)} \sum_{i \in SR(C_i)}^{|SR(C_i)|} u(i)$ is the average utility of all items $SR(C_i)$ that satisfy C_i. The utility of each item $u(i)$ is calculated based on the Multi-Attribute Utility Theory [7]. Due to space limit, more details can be referred to our earlier work [5].

The critique candidates with the highest ϕ scores are presented to the user, as the critique suggestions. Once the user selects a critique, the user's preferences (i.e., the weights and value functions placed on critiqued attributes) are accordingly refined. The system will then recommend a new item to the user in the next cycle. In the original preference-based organization system, the product that is with the highest utility as well as satisfying the user selected critique is recommended. However, it did not consider other users' history data, which motivates us to propose the following history-aware approach.

Similar Session Identification. Similar to related history-aware (also called experience-based) approaches [9,13,17], we also aim to incorporate *experiences* from past users, but the difference lies in the measurement of similar sessions. In our approach, each critiquing session is defined as a *sequence* of critiques (along with the critiqued items) made by a user during her/his interaction with the system.

Definition 2. *(Critiquing Session). The critiquing session, denoted as s_k, is a sequential vector with the recommended item $i_{x,k}$ and the compound critique $C_{x,k}$ of each cycle:*

$$s_k =< i_{1,k}, C_{1,k}; i_{2,k}, C_{2,k}; ...; i_{n,k}, C_{n,k} >$$

where $C_{x,k}$ is the compound critique made on item $i_{x,k}$, $C_{n,k} = \emptyset$ since the item $i_{n,k}$ is the final choice made by the user in that session, and n is the total number of critiquing cycles that the user consumes.

The purpose of similar session identification is then to identify whether two critiquing sessions are similar, for which the definition of proper similarity measure is crucial. In [9,13], *OverlapScore*, which gives the number of overlapping critiques between two sessions, was used to measure their similarity. [17] improved this metric by taking into account items' similarity as well:

$$Sim(s_i, s_j) = \beta \times ItemSim(s_i, s_j) + (1 - \beta) \times OverlapScore(s_i, s_j) \qquad (4)$$

where $ItemSim(s_i, s_j)$ is the average similarity between all items in two sessions s_i and s_j (i.e., $\frac{\sum_{i' \in s_i} \sum_{i^* \in s_j} Sim(i', i^*)}{|I'||I^*|})^2$, $OverlapScore(s_i, s_j)$ is the square of the number of overlapping critiques[3], and β was tuned as 0.75 in [17].

However, those similarity metrics do not consider the sequence of items/critiques that is embodied in the session. To illustrate this problem, we can take a look at the following example:

Example 1. Suppose the current critiquing session s_a is

$$s_a =< i_1, C_1; i_2, C_2 >$$

where $C_1 = \{c_1, c_2, c_3\}$ and $C_2 = \{c_4, c_5\}$. s_a can also be represented as:

$$s_a =< i_1, \{c_1, c_2, c_3\}; i_2, \{c_4, c_5\} >$$

We have two critiquing sessions S_b and S_c from other users' history data:

$$s_b =< i_1, \{c_1, c_2, c_3\}; i_2, \{c_4, c_5\}; i_3, \emptyset >$$

$$s_c =< i_1, \{c_1, c_2\}; i_2, \{c_3, c_4, c_5\}; i_4, \emptyset >$$

[2] $|I'|$ and $|I^*|$ are the numbers of items in s_i and s_j respectively (the finally accepted item is excluded).

[3] $OverlapScore(s_i, s_j) = [\sum_{c' \in s_i} \sum_{c^* \in s_j} match(c', c^*)]^2$; if $c' = c^*$, $match() = 1$, otherwise, $match() = 0$.

Obviously, the session s_b is more similar to session s_a than s_c (so we should recommend item i_3 for the current session), in that it contains the same critiques and items regarding the first two cycles, which are also with the same sequence, as in session s_a. However, if we adopt similarity metric Equation 4, the same similarity value will be obtained for sessions s_b and s_c, which is $\beta \times ItemSim(\{i_1, i_2\}, \{i_1, i_2\}) + (1 - \beta) \times 25$ (since there are 5 overlapping critiques: c_1, c_2, c_3, c_4 and c_5, so the square is 25). This problem is mainly caused by the fact that it neglects the sequential relationship between items/critiques. To address this problem, we propose a graph-based similarity measure. Specifically, a directed graph can be built for each critiquing session:

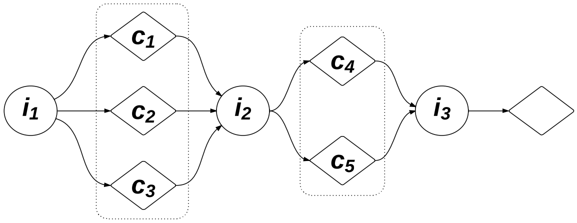

Fig. 2. The session graph built for $s_b =< i_1, \{c_1, c_2, c_3\}; i_2, \{c_4, c_5\}; i_3, \emptyset >$

Definition 3. *(Session Graph). The session graph for a critiquing session s_k, denoted as G_k, is in the form of two-tuple:*

$$G_k =< V_k, E_k > (V_k = I_k \cup C_k; E_k = E_k^{i \to c} \cup E_k^{i \leftarrow c})$$

where the vertex set $I_k = \cup_{j=1}^{n} i_{j,k}$ denoting all items in s_k, $C_k = \cup_{j=1}^{n} C_{j,k}$ which is the vertex set of all critiques contained in the session (note that each compound critique $C_{j,k}$ is composed of a set of unit critiques), and the edge set E_k includes two kinds of edge: $E_k^{i \to c}$ from an item vertex to a critique vertex (e.g., the edge from i_1 to c_1 in Figure 2), and $E_k^{i \leftarrow c}$ from a critique vertex to an item vertex (e.g., the edge from c_1 to i_2 in Figure 2).

Thus, it can be seen that the similarity metric Equation 4 takes only the graph's vertices into consideration while ignoring edges, that is why it can not distinguish sessions which are with the same vertices but different edges. The new similarity metric that we propose is given in Equation 5:

$$RSim(G_x, G_y) = \frac{|I_x \cap I_y|}{|I_x \cup I_y|} \cdot \frac{|C_x \cap C_y|}{|C_x \cup C_y|} \cdot \frac{|E_x^{i \to c} \cap E_y^{i \to c}|}{|E_x^{i \to c} \cup E_y^{i \to c}|} \cdot \frac{|E_x^{i \leftarrow c} \cap E_y^{i \leftarrow c}|}{|E_x^{i \leftarrow c} \cup E_y^{i \leftarrow c}|} \quad (5)$$

where G_x and G_y are two session graphs for critiquing sessions s_x and s_y respectively. The four considered factors are respectively item vertices, critique vertices, edges from critique to item, and edges from item to critique. To avoid zero result, we adopt $e^{RSim(G_x, G_y)}$ as the final similarity score. If we revisit Example 1 using Equation 5, $RSim(G_a, G_b) = 1$ and $RSim(G_a, G_c) = \frac{4}{9}$, which is consistent with our observation (i.e., session s_b is more similar to session s_a than s_c).

Item Recommendation. The next step is then to determine the item to be recommended for the current session. Similar to the idea suggested in [9], we consider all items contained in the most similar sessions as candidates for recommendation. However, instead of using $Compatibility(i_t, q)$ [4] as the ranking score [9], we define a R function to calculate an item's relevance to the current session:

$$R(i_t, q, S) = \operatorname*{argmax}_{\forall s_k \in S} e^{RSim(G(q), G(s_k^{(i_t)}))} \tag{6}$$

where q is the current critiquing session, S is the set of similar critiquing sessions from other users, $G(s_k^{(i_t)})$ is the session graph that starts from the start till the item i_t in session s_k (i.e., $s_k^{(i_t)} = s_k - \{i_t, C_t; ...; i_{n,k}, C_{n,k}\}$), and the maximal value $e^{RSim(G(q),G(s_k^{(i_t)}))}$ is taken as item i_t's ranking score. The item with the highest ranking score will hence be recommended to the user. At this point, either the user finds her/his target choice and thus terminates her/his interaction with the system, or s/he makes further critique in order to obtain more accurate recommendation in the subsequent cycle.

4 Experiment

Data Sets and Evaluation Metrics. Two public data sets were used for evaluating our proposed method. The first one is the car data set including 406 cars each characterized by 10 attributes (3 categorical and 7 numerical attributes) [14]. Another is the laptop data set which contains 836 laptop items each with 20 attributes (12 categorical and 8 numerical attributes) [2]. To measure the efficiency of our proposed method, we use the session length (i.e., the critiquing cycles consumed for reaching the target choice) as the metric, so as to identify whether it could reduce users' interaction cycles in the pre-condition that users do not need to compromise their decision accuracy (i.e., they are still able to find their target choice at the end). To perform simulation, we adopt the leave-one-out strategy that has been commonly used in related works [13,17]. To be specific, at one time, one item was randomly withdrawn from the dataset that is called "test item", and the item most similar to it is taken as the "target choice". A subset of attribute values of the test item are treated as the simulated user's initial preferences based on which the system will return the first recommendation (which is best matching to the user's initial preferences), and generate a set of compound critiques (by the method described in Section 3 "*Compound Critique Generation*"). The critique that is most compatible with the target choice is assumed being selected by the simulated user. Then, the critiquing sessions most similar to the current session will be determined (Section 3 "*Similar Session Identification*"), and the recommendation can then be decided for the next

[4] $Compatibility(i_t, q)$ is the number of satisfied critiques in the current session q for the item i_t (i.e., $Compatibility(i_t, q) = |\{c_i | satisfies(i_t, c_i), c_i \in q\}|$).

cycle (Section 3 "*Item Recommendation*"). The user's preferences will also be accordingly updated for generating new critiques in the next cycle. The process continues until the target choice is reached.

Compared Methods. We compared our method (shorted as *graph-based*) to the three most related ones (as mentioned in Section 2). See Table 1 for the summary of their main differences. We applied each method in both types of critiquing systems: *unit critiquing* and *compound critiquing*. The baseline unit critiquing approach is the standard one without considering other users' history data [3], and the baseline compound critiquing approach is the original preference-based organization method [5].

Table 1. Methods for experimental comparison

Method	**Short description**
Experience-based [13]	*OverlapScore* for similar session identification (Footnote 3)
NN-based[a] [9]	*Compatibility* for item recommendation (Footnote 4)
History-aware [17]	$Sim()$ for similar session identification (Equation 4)
Graph-based (our proposed method)	$RSim()$ for similar session identification (Equation 5) and $R()$ for item recommendation (Equation 6)

[a] NN is the abbreviation of 'nearest-neighbor'.

Results Analysis. The overall comparison between our proposed method and related ones can be found in Figure 3 (car dataset) and Figure 4 (laptop dataset). Note that we set $Size_{initial-pref} = 3$ (the initial preferences' size), $Size_{critiques} = 3$ (the number of unit critiques contained in each compound critique), and $Size_{base} = 406$ (the base size denoting the number of critiquing sessions from other users) for the overall comparison.

It can be seen that the results from both data sets show the similar trends: 1) all compound critiquing based methods take shorter session length than the corresponding unit critiquing based systems. The differences reach at significant level according to Student's t-test analysis ($p < 0.01$). 2) All history-aware methods are more effective than the baseline methods that do not consider other users' critiquing histories, which is valid in both unit critiquing ($p < 0.05$) and compound critiquing systems ($p < 0.01$). 3) Our proposed *graph-based* method achieves the best performance among all the compared methods for compound critiquing (e.g., with average 16.01% length reduction in car dataset and 14.25% length reduction in laptop dataset ($p < 0.05$)), which phenomenon is also valid in unit critiquing system. The results hence verify our hypothesis that considering sequential relationship between items/critiques in the critiquing session (as implemented in our *graph-based* method) can help identify similar sessions more accurately.

In Figure 5, we further show the methods' comparison in terms of recommendation accuracy on per cycle basis in an accumulated way (in the compound critiquing system with car dataset[5]). It shows that the *graph-based* method is more accurate than the others during each cycle. In addition, we can see all curves become convergent when the session length increases to 25.

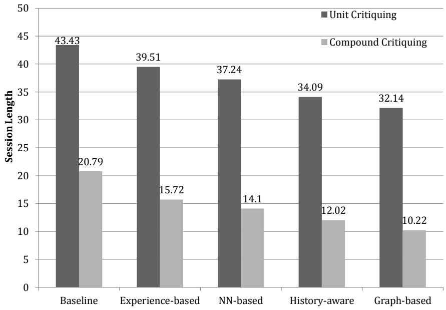

Fig. 3. Overall comparison among all methods (car dataset)

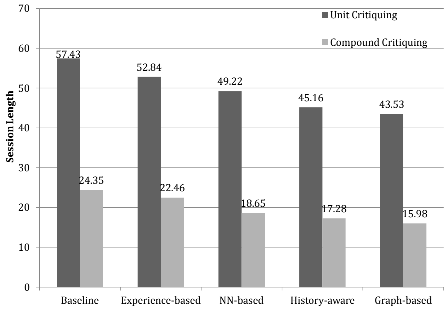

Fig. 4. Overall comparison among all methods (laptop dataset)

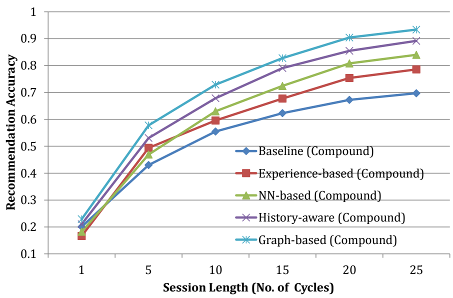

Fig. 5. Comparison w.r.t. recommendation accuracy on per cycle basis (car dataset)

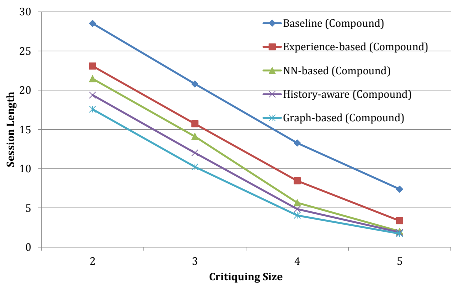

Fig. 6. Comparison w.r.t. critiquing size (car dataset)

Parameter Influence. As mentioned before, there are three main parameters: $Size_{critiques}$, $Size_{initial-pref}$, and $Size_{base}$. In Figure 6, we vary $Size_{critiques}$ from 2 to 5 (since according to [5], the maximal number of attributes contained in each compound critique should be no more than 5, in order to reduce information overload to users), with $Size_{initial-pref}$ and $Size_{base}$ respectively set as 3 and 406[6]. We can find that the session length decreases when $Size_{critiques}$ is increased, as the larger critiquing size will narrow down the product space. Furthermore, to see the effect of varying the initial preferences' size $Size_{initial-pref}$,

[5] The results in unit critiquing system and laptop data set show the similar trends, so the figures are not shown due to space limit.

[6] This is base size set in car dataset. In laptop dataset, it is set as 836.

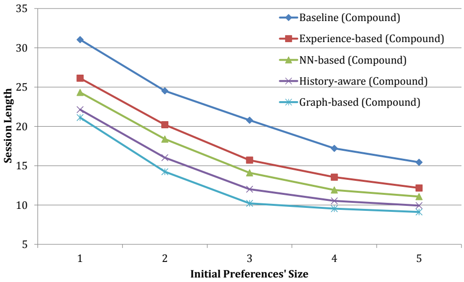

Fig. 7. Comparison w.r.t. initial preferences' size (car dataset)

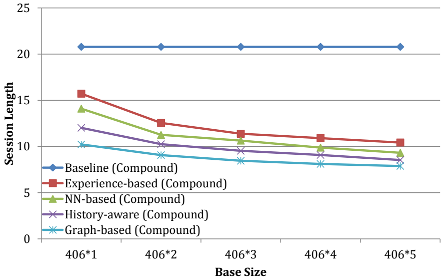

Fig. 8. Comparison w.r.t. base size (car dataset)

we set it's range from 1 to 5 (which covers all sizes used in [10,13]) and the other parameters with fixed values ($Size_{critiques} = 3$ and $Size_{base} = 406$). Similarly, it can be seen in Figure 7 that the session length decreases when $Size_{initial-pref}$ increases. Moreover, as shown in Figure 8, shorter session lengths for all methods are obtained when base size $Size_{base}$ is increased from 406 to 2436 in car dataset (when $Size_{critiques} = 3$ and $Size_{initial-pref} = 3$). This is mainly because the larger base size can provide more critiquing sessions for identifying similar ones to the current session. The chance of locating the target choice from these sessions would be higher. Besides, all of the figures show that the performance of *graph-based* method can always obtain the best result relative to the other approaches no matter which parameter is varied.

5 Conclusion

In this paper, we present a *collaborative compound critiquing* approach that not only generates compound critiques based on user preferences, but also locates recommended items from other users' critiquing histories. It models the critiquing session as a directed graph, and proposes a novel *graph-based* similarity measure to identify similar sessions. To understand the new approach's efficiency in saving users' interaction efforts, we conducted experiment on two data sets to compare it with three related history-aware (also called experience-based) critiquing approaches [9,13,17]. The experimental results show that our method achieves significantly higher efficiency than all of the compared methods. Moreover, it was found that the history-aware compound critiquing systems can take shorter session length than the corresponding unit critiquing systems, and they are also more effective than the baseline systems that do not involve other users' history data.

Thus, we believe that our collaborative compound critiquing approach can well improve the efficiency of existing critiquing based recommender systems in terms of reducing users' efforts, while still allowing them to reach target choice. In the future, we will validate these results through user evaluations. We will also investigate the algorithm's efficiency and scalability in larger-scale data sets.

Acknowledgements. We thank grants ECS/HKBU211912 and NSFC/ 61272365.

References

1. Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. IEEE Transactions on Knowledge and Data Engineering 17(6), 734–749 (2005)
2. Becerra, C., Gonzalez, F., Gelbukh, A.: Visualizable and explicable recommendations obtained from price estimation functions. In: Proc. ACM RecSys 2011, pp. 27–34 (2011)
3. Burke, R.D., Hammond, K.J., Yound, B.: The findme approach to assisted browsing. IEEE Expert 12(4), 32–40 (1997)
4. Chen, L.: User Decision Improvement and Trust Building in Product Recommender Systems. PhD thesis, Ecole Polytechnique Federale De Lausanne (EPFL), Lausanne, Switzerland (August 2008)
5. Chen, L., Pu, P.: Preference-based organization interfaces: Aiding user critiques in recommender systems. In: Conati, C., McCoy, K., Paliouras, G. (eds.) UM 2007. LNCS (LNAI), vol. 4511, pp. 77–86. Springer, Heidelberg (2007)
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. Keeney, R.L.: Decisions with Multiple Objectives: Preferences and Value Tradeoffs. Cambridge University Press (1993)
8. Linden, G., Hanks, S., Lesh, N.: Interactive assessment of user preference models: The automated travel assistant. In: Proc. UM 1997, pp. 67–68 (1997)
9. Mandl, M., Felfernig, A.: Improving the performance of unit critiquing. In: Masthoff, J., Mobasher, B., Desmarais, M.C., Nkambou, R. (eds.) UMAP 2012. LNCS, vol. 7379, pp. 176–187. Springer, Heidelberg (2012)
10. McCarthy, K., McGinty, L., Smyth, B.: Dynamic critiquing: An analysis of cognitive load. In: Proc. ICAICS 2005, pp. 19–28 (2005)
11. McCarthy, K., Reilly, J., McGinty, L., Smyth, B.: On the dynamic generation of compound critiques in conversational recommender systems. In: De Bra, P.M.E., Nejdl, W. (eds.) AH 2004. LNCS, vol. 3137, pp. 176–184. Springer, Heidelberg (2004)
12. Mccarthy, K., Reilly, J., Smyth, B., Mcginty, L.: Generating diverse compound critiques. Artificial Intelligence Review 24(3-4), 339–357 (2005)
13. McCarthy, K., Salem, Y., Smyth, B.: Experience-based critiquing: Reusing critiquing experiences to improve conversational recommendation. In: Bichindaritz, I., Montani, S. (eds.) ICCBR 2010. LNCS, vol. 6176, pp. 480–494. Springer, Heidelberg (2010)
14. Quinlan, J.R.: Combining instance-based and model-based learning. In: Proc. ICML 1993, pp. 236–243 (1993)
15. Reilly, J., McCarthy, K., McGinty, L., Smyth, B.: Dynamic critiquing. In: Funk, P., González Calero, P.A. (eds.) ECCBR 2004. LNCS (LNAI), vol. 3155, pp. 763–777. Springer, Heidelberg (2004)
16. Reilly, J., McCarthy, K., McGinty, L., Smyth, B.: Incremental critiquing. Knowledge-Based Systems 18(4), 143–151 (2005)
17. Salem, Y., Hong, J.: History-aware critiquing-based conversational recommendation. In: Proc. WWW 2013, pp. 63–64 (2013)
18. Zhang, J., Pu, P.: A comparative study of compound critique generation in conversational recommender systems. In: Wade, V.P., Ashman, H., Smyth, B. (eds.) AH 2006. LNCS, vol. 4018, pp. 234–243. Springer, Heidelberg (2006)