Preference-based Organization Interfaces: Aiding User Critiques in Recommender Systems

Li Chen and Pearl Pu

Human Computer Interaction Group, School of Computer and Communication Sciences
Swiss Federal Institute of Technology in Lausanne (EPFL)
CH-1015, Lausanne, Switzerland
{li.chen, pearl.pu}@epfl.ch

Abstract. Users' critiques to the current recommendation form a crucial feedback mechanism for refining their preference models and improving a system's accuracy in recommendations that may better interest the user. In this paper, we present a novel approach to assist users in making critiques according to their stated and potentially hidden preferences. This approach is derived from our previous work on critique generation and organization techniques. Based on a collection of real user data, we conducted an experiment to compare our approach with three existing critique generation systems. Results show that our preference-based organization interface achieves the highest level of prediction accuracy in suggesting users' intended critiques and recommendation accuracy in locating users' target choices. In addition, it can potentially most efficiently save real users' interaction effort in decision making.

Keywords: Recommender systems, user preference models, critique generation, organization, decision support, experiment.

1 Introduction

Recommender systems propose items that may interest a user. When it comes to suggesting decisions, such as which camera to buy, the ability to accurately recommend items that users truly want and reduce their effort in identifying the best choice is important. Decision accuracy and user effort are indeed two of the main factors influencing the design of product recommenders [8].

Many highly interactive recommender systems engage users in a conversational dialog in order to learn their preferences and use their feedback to improve the system's recommendation accuracy. Such interaction models have been referred as conversational recommenders, using both natural language models [14] and graphical user interfaces [2,12]. The main component of the interaction is that of example-and-critique. The system simulates an artificial salesperson that recommends example options based on a user's current preferences and then elicits his/her feedback in the form of critiques such as "I would like something cheaper" or "with faster processor speed". These critiques form the critical feedback mechanism to help the system improve its accuracy in predicting the user's needs in the next recommendation cycle.

Our previous work proved that intelligent critiquing support allows users to more effectively refine the quality of their preferences and improve their decision accuracy up to a higher degree, compared to the non critiquing-based system such as a ranked list [8,10]. We have also investigated and compared two approaches to help users adopt such critiquing support tools. One is the system-proposed critique generation technique that aims at proposing a set of critiques for users to choose, and another is the user self-motivated critiquing support which stimulates users to freely compose and combine critiques on their own [3]. A comparative user evaluation shows that users on average achieved higher confidence in choice and decision accuracy while being self-motivated to make critiques. However, some users still preferred the system-proposed critiques since they found it intuitive to use and potentially their decision process could be accelerated if the critiques closely matched the critiques they were prepared to make.

Motivated by these findings, we have been engaged in improving the critiquing-based recommender system mainly from two aspects. On the one hand, we have developed a hybrid critiquing system with the purpose of combining the two types of critiquing assistances and making them compensate for each other. The hybrid system was empirically shown to have potential to both effectively improve users' objective decision performance and promote their subjective perceptions [4].

On the other hand, given the limitation of traditional system-proposed critique generation approaches in predicting users' intended critiques (due to their purely data-driven selection mechanism), we have designed and implemented computation algorithms focusing on users' preferences. The critique generation method based on multi-attribute utility theory (MAUT) [5] was shown to more effectively stimulate users to apply the proposed critiques [16]. After testing different concrete interface designs with real users, we have further proposed the preference-based organization interface aimed at organizing the individual MAUT-based critiques into different categories and using the category titles (i.e. frequent critique patterns) as upper-level critique suggestions. This interface was demonstrated to more effectively promote users' trust in recommendations and increase their trusting intentions to return and save effort [9].

In this paper, we attempt to further evaluate the preference-based organization interface in terms of its actual accuracy in predicting critiques matching real users' intended criteria and in recommending products that are in fact users' target choices. Based on a collection of 54 real users' data, we compared our approach with three primary existing critique generation methods: the FindMe [2], dynamic critiquing system [11,12], and MAUT-based compound critiques [16].

2 Related Work

FindMe systems generate critiques according to their knowledge of the product domain. For example, the tweak application (also called assisted browsing) developed in one FindMe system (i.e. RentMe) allows users to critique the current recommended apartment by selecting one of the proposed simple tweaks (e.g. "cheaper", "bigger" and "nicer") [2]. However, since the critiques are pre-designed by the system, they may not reflect the current status of available products.

Table 1. Main differences between four system-proposed critique generation methods.

	Dynamic critiques	Critiques typical of the remaining products	Critiques adaptive to user preferences	Diversity among critiques and their contained products
Preference-based organization	✓	✓	✓	✓
MAUT-based compound critiques	✓	×	✓	×
Dynamic critiquing	✓	✓	×	Partially (only critiques)
FindMe	×	×	×	Partially (only critiques)

The dynamic critiquing method [11] and its successor, incremental critiquing [12], have been proposed mainly to automatically and dynamically generate compound critiques (e.g. "Different Manufacture, Lower Resolution and Cheaper" that can operate over multiple features simultaneously), by discovering the frequent sets of value differences between the current recommendation and remaining products based on Apriori algorithm [1]. Since a potentially large number of compound critiques would be produced by Apriori, they further filter all critiques using a threshold value favoring those critiques with lower support values ("support value" refers to the percentage of products that satisfy the critique). The dynamically generated critiques can also perform as explanations explaining to users the recommendation opportunities that exist in the remaining products [13].

However, the critique selection process purely based on support values indeed does not take into account users' preferences. It can only reveal "what the system can provide". For instance, the critique "Different Manufacture, Lower Resolution and Cheaper" is proposed if only there is a fewer percentage of products satisfying this critique. Even though the incremental dynamic critiquing method keeps a history of user previous critiques [12], the history only influences the computation of recommended products (i.e. requiring them compatible with the previous critique history as well as the current critique), not the process of critique generation.

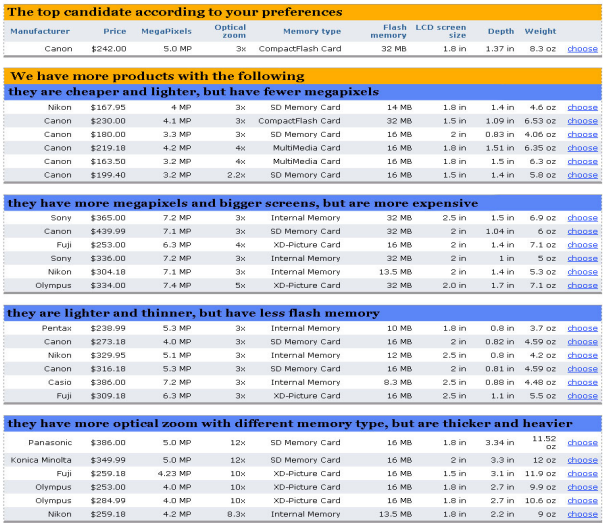
In order to respect user preferences in the proposed critiques, Zhang and Pu [16] have proposed an approach to adapting the generation of compound critiques to user preference models based on the multi-attribute utility theory (MAUT) [5]. During each recommendation cycle, several products best matching a user's current preferences will be computed and the detailed comparison of each of them with the top candidate will be presented as a compound critique. These preference-based compound critiques were shown to more likely match users' intended critiquing criteria. However, relative to the dynamic critiquing approach, this method is limited in exposing remaining recommendation opportunities since each MAUT-based compound critique only corresponds to one product. In addition, it does not provide diversity among critiques. From real users' point of view, each critique also contains too many attributes so as to likely cause information overload.

With the aim of keeping these approaches' advantages while compensating for their limitations, we have further developed the preference-based organization interface. It was designed not only dynamically generating critiques adaptive to users'

current preferences and potential needs, but also applying the data mining technique to produce representative compound critiques typical of the remaining data set. In addition, the critiques and their contained products are diversified so as to potentially assist users in refining and accumulating their preferences more effectively. Table 1 summarizes the main differences between the preference-based organization technique and other system-proposed critique generation methods.

3 Preference-based Organization Interface

To derive effective design principles for the preference-based organization interface, we previously designed more than 13 paper prototypes and tested them with real users in form of pilot studies and interviews (see details in [9]). Four primary principles were derived covering almost all design dimensions, such as proposing improvements and compromises in the critique using conversational language (principle 1), keeping the number of tradeoff attributes in the critique under five to avoid information overload (principle 2), including actual products (up to six) under the critique (principle 3), and diversifying the proposed critiques and their contained products (principle 4) (the critique was termed as "category title" in [9]).



The top candidate according to your preferences

Manufacturer	Price	MegaPixels	Optical zoom	Memory type	Flash memory	LCD screen size	Depth	Weight	
Canon	$242.00	5.0 MP	3x	CompactFlash Card	32 MB	1.8 in	1.37 in	8.3 oz	choose

We have more products with the following

they are cheaper and lighter, but have fewer megapixels

Nikon	$167.95	4 MP	3x	SD Memory Card	14 MB	1.8 in	1.4 in	4.6 oz	choose
Canon	$230.00	4.1 MP	3x	CompactFlash Card	32 MB	1.5 in	1.09 in	6.53 oz	choose
Canon	$180.00	3.3 MP	3x	SD Memory Card	16 MB	2 in	0.83 in	4.06 oz	choose
Canon	$219.18	4.2 MP	4x	MultiMedia Card	16 MB	1.8 in	1.51 in	6.35 oz	choose
Canon	$163.50	3.2 MP	4x	MultiMedia Card	16 MB	1.8 in	1.5 in	6.3 oz	choose
Canon	$199.40	3.2 MP	2.2x	SD Memory Card	16 MB	1.5 in	1.4 in	5.8 oz	choose

they have more megapixels and bigger screens, but are more expensive

Sony	$365.00	7.2 MP	3x	Internal Memory	32 MB	2.5 in	1.5 in	6.9 oz	choose
Canon	$439.99	7.1 MP	3x	SD Memory Card	32 MB	2 in	1.04 in	6 oz	choose
Fuji	$253.00	6.3 MP	4x	XD-Picture Card	16 MB	2 in	1.4 in	7.1 oz	choose
Sony	$336.00	7.2 MP	3x	Internal Memory	32 MB	2 in	1 in	5 oz	choose
Nikon	$304.18	7.1 MP	3x	Internal Memory	13.5 MB	2 in	1.4 in	5.3 oz	choose
Olympus	$334.00	7.4 MP	5x	XD-Picture Card	32 MB	2.0 in	1.7 in	7.1 oz	choose

they are lighter and thinner, but have less flash memory

Pentax	$238.99	5.3 MP	3x	Internal Memory	10 MB	1.8 in	0.8 in	3.7 oz	choose
Canon	$273.18	4.0 MP	3x	SD Memory Card	16 MB	2 in	0.82 in	4.59 oz	choose
Nikon	$329.95	5.1 MP	3x	Internal Memory	12 MB	2.5 in	0.8 in	4.2 oz	choose
Canon	$316.18	5.3 MP	3x	SD Memory Card	16 MB	2 in	0.81 in	4.59 oz	choose
Casio	$386.00	7.2 MP	3x	Internal Memory	8.3 MB	2.5 in	0.88 in	4.48 oz	choose
Fuji	$309.18	6.3 MP	3x	XD-Picture Card	16 MB	2.5 in	1.1 in	5.5 oz	choose

they have more optical zoom with different memory type, but are thicker and heavier

Panasonic	$386.00	5.0 MP	12x	SD Memory Card	16 MB	1.8 in	3.34 in	11.52 oz	choose
Konica Minolta	$349.99	5.0 MP	12x	SD Memory Card	16 MB	2 in	3.3 in	12 oz	choose
Fuji	$259.18	4.23 MP	10x	XD-Picture Card	16 MB	1.5 in	3.1 in	11.9 oz	choose
Olympus	$253.00	4.0 MP	10x	XD-Picture Card	16 MB	1.8 in	2.7 in	9.9 oz	choose
Olympus	$284.99	4.0 MP	10x	XD-Picture Card	16 MB	1.8 in	2.7 in	10.6 oz	choose
Nikon	$259.18	4.2 MP	8.3x	Internal Memory	13.5 MB	1.8 in	2.2 in	9 oz	choose

Fig. 1. The preference-based organization interface.

We have accordingly developed an algorithm to optimize the objectives corresponding to these principles (see Fig. 1 of a resulting interface). Note that in our interface design, multiple products that satisfy the proposed critique are recommended simultaneously, rather than only one product returned (once a critique is picked) in the traditional system-proposed critiquing interfaces [2,12,16]. This interface was in fact favored by most of interviewed users since it could potentially save their interaction effort and give them higher control over the process of choice making. The following lists the main characteristics of our algorithm as how it models and

incrementally refines user preferences, and how critiques are generated typical of the remaining products and selected adaptive to user preferences and potential needs.

Model user preferences based on MAUT. We represent the user preferences over all products as a weighted additive form of value functions according to the multi-attribute utility theory (MAUT) [5,16]. This MAUT-based user model is inherently in accordance with the most normal and compensatory decision strategy, the weighted additive rule (WADD) that resolves conflicting values explicitly by considering tradeoffs [7]. Formally, the preference model is a pair ($\{V_1,...,V_n\}$, $\{w_1,...,w_n\}$) where V_i is the value function for each attribute A_i, and w_i is the relative importance (i.e. weight) of A_i. The utility of each product ($\langle a_1, a_2, ..., a_n \rangle$) can be hence calculated as:

$$U(\langle a_1, a_2, ..., a_n \rangle) = \sum_{i=1}^{n} w_i V_i(a_i) \quad (1)$$

Suggest unstated preferences in critiques. Giving user suggestions on unstated preferences was demonstrated to likely stimulate preferences expression and improve users' decision accuracy [15]. Thus, while generating the critique pattern of each remaining product by comparing it with the current recommendation (i.e. the top candidate), we assign default tradeoff properties (i.e. *improved* or *compromised*) to these features without explicit stated preferences. For example, if a user does not specify any preference on the notebook's processor speed, we will assign *improved* (if faster) or *compromised* (if slower) to the compared product's processor speed. We believe that the proposed critiques with suggested preferences could help users learn more knowledge about the product domain and potentially stimulate them to expose more hidden preferences.

Produce critiques typical of the remaining products. In our algorithm, each product (except the top candidate) will be turned into a tradeoff vector (i.e. critique pattern) comprising a set of (*attribute*, *tradeoff*) pairs. The *tradeoff* property is determined by the user's stated preference or our suggested direction. More concretely, it indicates whether the *attribute* of the product is *improved* (denoted as ↑) or *compromised* (denoted as ↓) compared to the same *attribute* of the top candidate. For example, a notebook's tradeoff vector can be represented as {(price, ↓), (processor speed, ↑), (hard drive size, ↑), (display size, ↓), (weight, ↑)}.

We then apply the Apriori algorithm to discover the recurring and representative subsets of (*attribute*, *tradeoff*) pairs within these tradeoff vectors (the discovered subset is called a "compound critique" or "category title" [9]). The reason of applying Apriori is due to its efficiency and popularity in mining associate rules among features [1]. Additionally, it provides various parameters enabling us to control the number of attributes involved in each critique and the percentage of products each critique contains so as to satisfy our design principles (principle 2 and 3).

Thus, at this point, all remaining products can be organized into different categories and each category be represented by a compound critique (e.g. "cheaper and lighter but lower processor speed") indicating the similar tradeoff properties of products that this category contains (principle 1).

Favor critiques with higher tradeoff utilities. The Apriori algorithm will potentially produce a large amount of critiques since a product can belong to more than one

category given that it has different subsets of tradeoff properties shared by other groups of products. It then comes to the problem of how to select the most prominent critiques presented to users. In stead of simply selecting critiques with lower support values as the dynamic critiquing method does [11,12], we focus on using users' preferences and their potential needs to choose critiques. More specifically, all critiques are ranked according to their tradeoff utilities (i.e. gains vs. losses relative to the top candidate) in terms of both the critiques themselves and their contained products:

$$TradeoffUtility(C) = (\sum_{i=1}^{|C|} w(attribute_i) \times tradeoff_i) \times (\frac{1}{|SR(C)|} \sum_{r \in SR(C)}^{|SR(C)|} U(r)) \quad (2)$$

where C denotes the critique as a set of (*attribute*, *tradeoff*) pairs, and *SR(C)* denotes the set of products that satisfy C. Therefore, according to the user's stated preferences and our suggestions on his/her potential needs, $\sum_{i=1}^{|C|} w(attribute_i) \times tradeoff_i$ computes the weighted sum of tradeoff properties represented by C ($w(attribute_i)$ is the weight of $attribute_i$; $tradeoff_i$ is default set as 0.75 if *improved*, or 0.25 if *compromised*, since *improved* attributes are in nature more valuable than *compromised* ones). $\frac{1}{|SR(C)|} \sum_{r \in SR(C)}^{|SR(C)|} U(r)$ is the average utility (see formula (1)) of all the products contained by C.

Diversify proposed critiques and their contained products. To further diversify the proposed critiques to increase their suggestion power since similar items are limited to add much useful values to users [6] (principle 4), we multiply the tradeoff utility of each critique by a diversity degree:

$$F(C) = TradeoffUtility(C) \times Diversity(C, SC) \quad (3)$$

where *SC* denotes the set of critiques so far selected. The first proposed critique is hence the critique with the highest tradeoff utility, and the subsequent critique is selected if it has the highest value of $F(C)$ in the remaining non-selected critiques. The selection process ends when the desired k critiques have been determined.

The diversity degree of C is concretely calculated as the minimal local diversity of C with each critique C_i in the *SC* set. The local diversity of two critiques is defined by two factors: the diversity between critiques themselves (i.e. C and C_i) and the diversity between their contained products (i.e. *SR(C)* and $SR(C_i)$):

$$Diversity(C, SC) = \min_{C_i \in SC} ((1 - \frac{|C \cap C_i|}{|C|}) \times (1 - \frac{|SR(C) \cap SR(C_i)|}{|SR(C)|})) \quad (4)$$

Incrementally refine user preferences. After a user has selected one of the proposed critiques and a new reference product from the set of products that satisfy the selected critique, his/her preferences will be accordingly refined for the computation of critiques in the next cycle. More concretely, the weight (i.e. relative importance) of *improved* attribute(s) that appears in the selected critique will be increased by β, and the weight of *compromised* one(s) will be decreased by β (β = 0.25). All attributes' preferred values will be also updated based on the new reference product's values.

4 Experimental Results

4.1 Materials and Procedure

The goal of this experiment is to evaluate the performance of the preference-based organization interface in terms of its accuracy in predicting critiques that users are likely to make and in recommending products that are targeted by users. We particularly compared our system with three primary existing critique generation approaches: MAUT-based compound critiques [16], dynamic critiquing [11,12], and FindMe [2].

As a matter of fact, few earlier works have empirically measured the prediction accuracy of their algorithms in suggesting critiques. In respect of their simulation experiments, a product randomly chosen from the database was used to determine a simulated user's target choice and his/her initial preferences [11,12,16].

The difference of our experiment is that it was based on a collection of real users' data so that it can potentially more realistically and accurately reveal the system's actual critique prediction accuracy and recommendation accuracy. The data has been concretely collected from a series of previous user studies where users were instructed to identify their truly intended critiquing criteria in the user self-motivated critiquing interface [3]. So far, 54 real users' records have been accumulated (with around 1500 data points), half of them asked to find a favorite digital camera (64 products, 8 main features) and the other half for a tablet PC (55 products, 10 main features). Each record includes the real user's initial preferences (i.e. a set of <attribute preferred value, attribute weight> pairs), the product he/she selected for critiquing and his/her self-motivated critiquing criteria (i.e. attributes to be *improved* or *compromised*) during each critiquing cycle, the total interaction cycles he/she consumed, and his/her target choice which was determined after he/she reviewed all products in an offline setting.

In the beginning of our experiment, each real user's initial preferences were first entered in the evaluated system. The system then proposed k critiques (k = 4), and the critique most matching the real user's intended critiquing criteria during that cycle was selected. Then, among the set of n recommended products (n = 6) that satisfy the selected critique, the product most similar to the actual product picked in that cycle was used for the next round of critique generation. This process ended when the corresponding real user stopped. That is, if a real user took three critiquing cycles to locate his/her final choice, he/she would also end after three cycles in our experiment.

4.2 Measured Variables and Results

4.2.1 Critique Prediction Accuracy

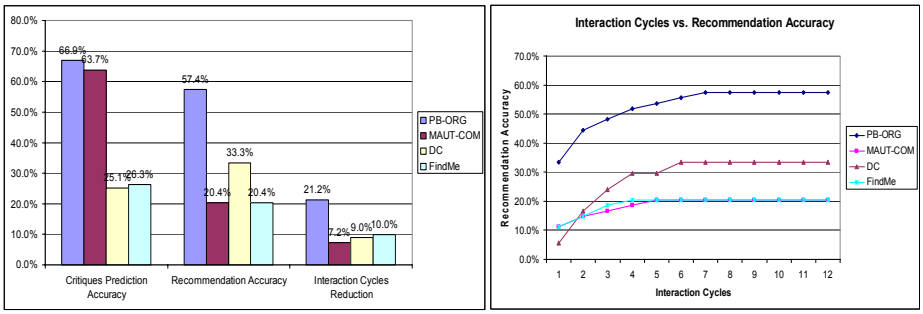
The critique prediction accuracy for each user is defined as the average matching degree between his/her self-motivated critiquing criteria and the most matching system-proposed critique of each cycle (see formula (5)). A higher matching degree infers that the corresponding critique generation algorithm can likely be more accurately predicting the critiques that real users intend to make.

$$PredictionRate(user_i) = \frac{1}{NumCycle} \sum_{j=1}^{NumCycle} \max_{c \in C_j} \left(\frac{\alpha \times NumImproveMatch(c) + (1-\alpha) \times NumCompromiseMatch(c)}{\alpha \times NumImprove(t) + (1-\alpha) \times NumCompromise(t)} \right) \quad \textbf{(5)}$$

where C_j represents the set of system-proposed critiques during the j^{th} cycle, *NumImprove(t)* is the number of improved attributes in the real user's critique

(denoted as t) during that cycle, and *NumCompromise(t)* is the number of compromised attributes. *NumImproveMatch(c)* denotes the number of improved attributes that appear in both the proposed critique (i.e. c) and the user's actual critique, and *NumCompromiseMatch(c)* is the number of matched compromised attributes (α= 0.75, since users likely want more accurate matching on the improved attributes).

The experimental results show that both the user preferences based critique generation approaches, the preference-based organization (henceforth PB-ORG) and MAUT-based compound critiques (henceforth MAUT-COM), achieve relatively higher success rate (respectively 66.9% and 63.7%) in predicting the critiques users actually made, compared to the dynamic critiquing method (henceforth DC) and FindMe approach (F = 94.620, p < 0.001; see Fig. 2 (a)). The PB-ORG is even slightly better than MAUT-COM. It therefore implies that when the proposed critiques can be well adaptive to the user's changing preferences and his/her potential needs, the user will likely more frequently apply them in the real situation.



(a) (b)

Fig. 2. Experimental comparison of four critique generation algorithms.

4.2.2 Recommendation Accuracy

In addition to evaluate the system's ability in predicting critiques, we also measured its recommendation accuracy as how likely users' target choices could have been located in the recommended products once the critique was made.

$$RecommendationAccuracy = \frac{1}{NumUsers} \sum_{i=1}^{NumUsers} FindTarget(target_i, \sum_{j=1}^{NumCycle(u_i)} RC_j(u_i)) \quad (6)$$

In this formula, $RC_j(u_i)$ denotes the set of recommended products that satisfy the selected critique during the j^{th} cycle for the user u_i. If the user's target choice (denoted as $target_i$) appears in any $RC_j(u_i)$ set, *FindTarget* is equal to 1, otherwise *FindTarget* is 0. The higher overall recommendation accuracy hence represents the larger proportion of users whose target choice appeared at least in one recommendation cycle, inferring that the corresponding system can likely more effectively recommend the target choice to real users during their acceptable critiquing cycles.

The experiment indicates that PB-ORG achieves the highest recommendation accuracy (57.4%) compared to the other systems (F = 8.171, p < 0.001; see Fig. 2 (a)). Fig. 2 (b) further illustrates the comparison of recommendation accuracy on a per cycle basis in an accumulated way (the maximal number of interaction cycles is 12). It is worth noting that although MAUT-COM obtains relatively higher critique

prediction accuracy compared to DC and FindMe, it is rather limited to recommend accurate products. In fact, regarding the recommendation accuracy, the best two approaches (PB-ORG and DC) are both based on the organization technique, and PB-ORG performs much better than DC likely due to its user preferences based selection mechanism. Therefore, PB-ORG is proven not only being most accurate at suggesting critiques that real users intended to make, but also most accurate at recommending products that were targeted by real users.

4.2.3 Interaction Effort Reduction

It is then interesting to know how effectively the system could potentially reduce real users' objective effort in locating their target choice. This was concretely measured as the percentage of cycles the average user could have saved to make the choice relative to the cycles he/she actually consumed in the self-motivated critiquing condition:

$$EffortReduction = \frac{1}{NumUsers}\left(\sum_{i=1}^{NumUsers} \frac{actualCycle_i - targetCycle_i}{actualCycle_i}\right) \quad (7)$$

where $actualCycle_i$ denotes the number of cycles the corresponding real user consumed and $targetCycle_i$ denotes the number of cycles until his/her target choice first appeared in the products recommended by the system. For the user whose target choice did not appear in any recommendations, his/her effort reduction is 0.

In terms of this aspect, PB-ORG again shows the best result ($F = 4.506$, $p < 0.01$; see Fig. 2 (a)). More specifically, the simulated user can on average save over 21.2% of their critiquing cycles while using the preference-based organization algorithm (vs. 7.2% with MAUT-COM, 8.95% with DC and 9.96% with FindMe). This finding implies that the preference-based organization interface can potentially enable real users to more efficiently target their best choice, not only relative to the user self-motivated critiquing system (where the *actualCycle* was consumed), but also compared to the other system-proposed critiquing systems.

5 Conclusion

In this paper, we described a new approach to generating proposed critiques based on users' preferences. The preference-based organization method computes critiques not only with MAUT-based user preference models but also with additional considerations such as classification and diversification. It organizes the critiques so as to identify the most prominent and representative critiques in the set of eligible critiques. To understand the new approach's accuracy in predicting critiques that users are likely to make and furthermore its accuracy in recommending products that are targeted by real users, we conducted an experiment to compare it with three primary existing critique generation approaches based on a collection of 54 real users' data. The experimental results show that both preference-based critique generation algorithms (PB-ORG and MAUT-based compound critiques [16]) achieve significantly higher critique prediction accuracy (above 60%), compared to the dynamic critiquing method (purely data-driven critique selection) [11,12] and the FindMe approach (pre-designed critiques) [2]. In addition, PB-ORG is most accurate

at recommending users' target choice (57.4%), while potentially requiring users to consume the least amount of interaction effort (by saving up to 22% critiquing cycles).

Thus, as a conclusion of our previous and current work, we believe that the preference-based organization interface can be well combined with the user self-motivated critiquing support [4] to maximally improve users' decision accuracy while demanding a low amount of users' objective and subjective effort. In addition, such hybrid critiquing system is likely to promote users' high subject opinions (i.e. trust and decision confidence) given that users can not only feel in control of their preference refinement process with the aid of user self-motivated critiquing support, but also have the opportunity to learn the remaining recommendation opportunities and accelerate their decision process in the preference-based organization interface. In the future, we will further verify these results via real user trials. We will also establish a more consolidated and sharable set of ground truth with more real users' data for the performance measurements of various recommender systems.

References

1. Agrawal, R., Imielinski, T., Swami, A.: Mining Association Rules between Sets of Items in Large Databases. In Proc. ACM SIGMOD (1993) 207–216.
2. Burke, R.D., Hammond, K.J., Young, B.C.: The FindMe Approach to Assisted Browsing. IEEE Expert: Intelligent Systems and Their Applications, Vol. 12(4) (1997) 32-40.
3. Chen, L., Pu, P.: Evaluating Critiquing-based Recommender Agents. In Proc. 21st AAAI (2006) 157-162.
4. Chen, L., Pu, P.: Hybrid Critiquing-based Recommender Systems. In Proc. IUI (2007) 22-31.
5. Keeney, R., Raiffa, H.: Decisions with Multiple Objectives: Preferences and Value Tradeoffs. Cambridge University Press (1976).
6. McGinty, L., Smyth, B.: On the Role of Diversity in Conversational Recommender Systems. In Proc. 5th ICCBR (2003) 276-290.
7. Payne, J.W., Bettman, J.R., Johnson, E.J.: The Adaptive Decision Maker. Cambridge University Press (1993).
8. Pu, P., Chen, L.: Integrating Tradeoff Support in Product Search Tools for e-commerce Sites. In Proc. 6th ACM EC (2005) 269-278
9. Pu, P., Chen, L.: Trust Building with Explanation Interfaces. In Proc. IUI (2006) 93-100.
10. Pu, P., Kumar, P.: Evaluating Example-Based Search Tools. In Proc. 5th ACM EC (2004) 208-217.
11. Reilly, J., McCarthy, K., McGinty, L., Smyth, B.: Dynamic Critiquing. In Proc. 7th ECCBR (2004) 763-777.
12. Reilly, J., McCarthy, K., McGinty, L., Smyth, B.: Incremental Critiquing. In Proc. 24th SGAI International Conference on Innovative Techniques and Applications of Artificial Intelligence (2004) 101-114.
13. Reilly, J., McCarthy, K., McGinty, L., Smyth, B.: Explaining Compound Critiques. Artificial Intelligence Review, Vol. 24 (2) (2005).
14. Thompson, C.A., Goker, M.H., Langley. P.: A Personalized System for Conversational Recommendations. Journal of Artificial Intelligence Research 21 (2004) 393-428.
15. Viappiani, P., Faltings, B., Pu, P.: Preference-based Search using Example-Critiquing with Suggestions. To appear in Journal of Artificial Intelligence Research (2007).
16. Zhang, J., Pu, P.: A Comparative Study of Compound Critique Generation in Conversational Recommender Systems. In Proc. AH (2006) 234-243.