A Comparison of Two Compound Critiquing Systems

James Reilly [1], *Jiyong Zhang* [2], *Lorraine McGinty* [1], *Pearl Pu* [2] and *Barry Smyth* [1]

[1] Adaptive Information Cluster
School of Computer Science & Informatics
UCD Dublin, Ireland

[2] Human Computer Interaction Group
Swiss Federal Institute of Technology (EPFL),
CH-1015, Lausanne, Switzerland

ABSTRACT

Compound critiques allow users to simultaneously express directional preferences over several product attributes. Presenting the user with compound critiques is not a new idea. The original Find-Me Systems (e.g., Car Navigator) showed static compound critiques; they didn't change irrespective of user preferences or the product availability. Recently, a number of techniques for *dynamically* generating compound critiques have been proposed. While these techniques have been evaluated in isolation, to date no direct comparison of these (in terms of their interfacing characteristics and recommendation performance) has been reported. Motivated by this, our research groups have come together to carry out this comparison for the approaches we each take. The user study platform that we have developed facilitates the comparison of various critiquing based recommenders. In this paper we report the first set of results from a comprehensive real-user evaluation of two dynamic compound critique systems using this evaluation platform.

ACM Classification: H.1.2 [Models and Principles]: User /Machine Systems – Human factors, Human information processing; H.5.2 [Information Interfaces and Presentation]: User Interfaces – Evaluation/methodology.

General terms: Human Factors, Performance, Experimentation

Keywords: user study, compound critiquing, recommender system

INTRODUCTION

Conversational recommender systems help prospective buyers quickly navigate to suitable products by facilitating the incremental construction of a more accurate picture of their requirements through a series of recommendation interactions [2]. In the course of each recommendation cycle a user is afforded the opportunity to provide feedback on suggestions and a conversational recommender will typically uses this feedback to influence subsequent recommendation retrievals. There are a variety of feedback modes that such a recommender could use (see [4] for more details). However, in this work we will concentrate on the well-known critiquing-based systems [2]. Put simply, a critique allows the user to constrain a particular product feature without requiring them to provide a specific value. For instance, when interacting with a PC recommender a user might indicate that they are looking for a 'cheaper' computer by critiquing the *Price* feature of a presented example. Thus, the standard approach to critiquing focusses on so-called *unit-critiques* that constrain a single feature at a time. Recently, researchers have explored the possibility of critiquing multiple features simultaneously in order to facilitate faster progression through the product-space.

In this paper we will review and compare two very different approaches to the dynamic generation of compound critiques. The first approach, Apriori, uses a data-mining algorithm to discover patterns in the types of products remaining, then converts these patterns into compound critiques. The second approach, MAUT, takes a utility-based decision theory approach to identify the most suitable products for users and converts these into a compound critique representation. Prompted by feedback from peers to both of our research groups, we set out to design a suitable evaluation platform that could be used to comparatively evaluate these techniques in a realistic product recommender. Ideally, this exercise would allow us to learn how to improve and/or look at ways of marrying ideas from both approaches. In this paper we summarize our initial findings from a first real-user trial using this evaluation platform which implements both of the compound critiquing approaches (further described below).

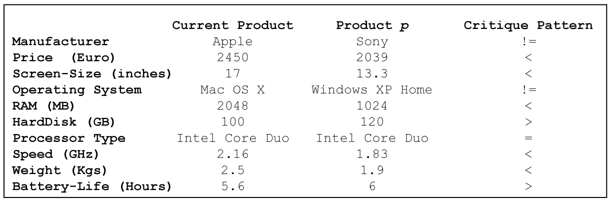
APPROACH 1: APRIORI

One strategy for dynamically generating compound critiques, proposed in [5], discovers feature patterns that are common to remaining products on every recommendation cycle. Essentially, each compound critique describes a set of products in terms of the feature characteristics they have in common. For example, in Figure 2 we see an example of a compound critique for *Faster CPU* and a *Larger Hard-Disk*. By clicking on this the user narrows the focus of the recommender to only those products that satisfy these feature preferences. The Apriori data-mining algorithm [1] is used to quickly discover these patterns and convert them into compound critiques on each recommendation cycle.

The first step involves *generating critique patterns* for each of the remaining product options in relation to the currently presented example. Figure 1 shows how a critique pattern for

IUI'07, January 28–31, 2007, Honolulu, Hawaii, USA.

a sample product p differs from the current recommendation for its individual feature critiques. For example, the critique pattern shown includes a "$<$" critique for Price— we will refer to this as $[Price <]$—because the comparison laptop is cheaper than the current recommendation. The next step involves *mining compound critiques* by using the Apriori algorithm [1] to identify groups of recurring unit critiques; we might expect to find the co-occurrence of unit critiques like $[ProcessorSpeed >]$ infers $[Price >]$. Apriori returns lists of compound critiques of the form $\{[ProcessorSpeed >], [Price >]\}$ along with their *support* values (i.e., the % of critique patterns for which the compound critique holds). It is not practical to present large numbers of different compound critiques as user-feedback options in each cycle. For this reason, a filtering strategy is used to select the k most useful critiques for presentation based on their support values. Importantly, compound critiques with low support values eliminate many more products from consideration if chosen.



	Current Product	Product *p*	Critique Pattern
Manufacturer	Apple	Sony	!=
Price (Euro)	2450	2039	<
Screen-Size (inches)	17	13.3	<
Operating System	Mac OS X	Windows XP Home	!=
RAM (MB)	2048	1024	<
HardDisk (GB)	100	120	>
Processor Type	Intel Core Duo	Intel Core Duo	=
Speed (GHz)	2.16	1.83	<
Weight (Kgs)	2.5	1.9	<
Battery-Life (Hours)	5.6	6	>

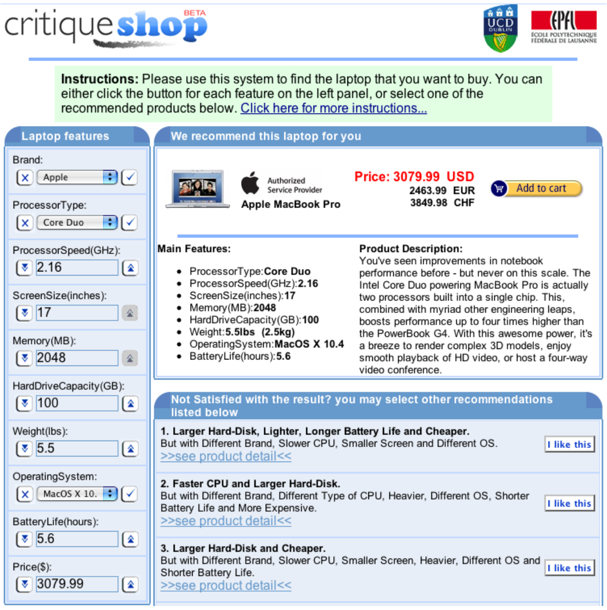
Figure 1: Generating a critique pattern.

The final step involves constructing a model of user preferences from the critiques specified so far. Importantly, users are not always consistent in the feedback they provide, so the aim of the model is to resolve any preference conflicts that may arise as the session proceeds. Put simply, when making a recommendation, the system computes a compatibility score for every product (informed by their critiquing history), and ranks them accordingly. This *incremental critiquing* approach [6] has been shown to deliver significant benefits in terms of recommendation quality and efficiency in prior evaluations.

APPROACH 2: MAUT

Using multi-attribute utility theory (MAUT) [3], Zhang and Pu have [7] developed a very different technique for dynamically generating compound critiques. In this approach, the system maintains a preference model based on user feedback. Before generating compound critiques, this model computes utility scores for the remaining products, ranks them accordingly, and compares the top ranking products to the current recommendation. A key difference in this approach to critique generation is that individual compound critiques relate to one product option only (as opposed to a set), whereby the compound critique serves as an alternative representation to describe that product.

Instead of mining the critiques directly from the product set based on the Apriori algorithm, the MAUT approach first determines top k products with maximal utilities, and then for each of the top k products, the corresponding critique-pattern is generated by comparing it with the current reference product in the same way as described in the previous section. The critique-patterns are then converted into natural language for presentation to the users.


critique shop BETA
UCD DUBLIN
EPFL ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE
Authorized Service Provider
Add to cart
I like this
I like this
I like this

Instructions: Please use this system to find the laptop that you want to buy. You can either click the button for each feature on the left panel, or select one of the recommended products below. Click here for more instructions...

Laptop features

Brand: Apple

ProcessorType: Core Duo

ProcessorSpeed(GHz): 2.16

ScreenSize(inches): 17

Memory(MB): 2048

HardDriveCapacity(GB): 100

Weight(lbs): 5.5

OperatingSystem: MacOS X 10.

BatteryLife(hours): 5.6

Price($): 3079.99

We recommend this laptop for you

Apple MacBook Pro

Price: 3079.99 USD
2463.99 EUR
3849.98 CHF

Main Features:

- ProcessorType:**Core Duo**
- ProcessorSpeed(GHz):**2.16**
- ScreenSize(inches):**17**
- Memory(MB):**2048**
- HardDriveCapacity(GB):**100**
- Weight:**5.5lbs (2.5kg)**
- OperatingSystem:**MacOS X 10.4**
- BatteryLife(hours):**5.6**

Product Description:
You've seen improvements in notebook performance before - but never on this scale. The Intel Core Duo powering MacBook Pro is actually two processors built into a single chip. This, combined with myriad other engineering leaps, boosts performance up to four times higher than the PowerBook G4. With this awesome power, it's a breeze to render complex 3D models, enjoy smooth playback of HD video, or host a four-way video conference.

Not Satisfied with the result? you may select other recommendations listed below

1. Larger Hard-Disk, Lighter, Longer Battery Life and Cheaper.
But with Different Brand, Slower CPU, Smaller Screen and Different OS.
>>see product detail<<

2. Faster CPU and Larger Hard-Disk.
But with Different Brand, Different Type of CPU, Heavier, Different OS, Shorter Battery Life and More Expensive.
>>see product detail<<

3. Larger Hard-Disk and Cheaper.
But with Different Brand, Slower CPU, Smaller Screen, Heavier, Different OS and Shorter Battery Life.
>>see product detail<<

Figure 2: Screenshot of system (MAUT interface).

When the user selects a compound critique, the corresponding critique product is assigned as the new reference product and the user's preference model is updated based on this critique selection. For each attribute, the attribute value of the new reference product is assigned as the preference value, and the weight of each attribute is adaptively adjusted according to the difference between the old preference value and the new preference value. Based on the new reference product and the new user preference model, the system is able to recommend another set of compound critiques for the users to critique until they find a suitable product.

EVALUATION

Previous studies have highlighted the effectiveness of dynamic compound critiques over unit-critiques through offline simulations and real user trials. Apriori-generated compound critiques have been shown to help deliver significant reductions in session-length, and users have also reported greater satisfaction when using them [4]. In a simulated environment, MAUT-generated compound critiques have shown further improvements in terms recommendation efficiency [7]. However, the absence of a direct comparison of these techniques in a real-user setting has meant that we have been unable to comment on their operational similarities/differences. Accordingly, we have designed a trial that asks users to compare two systems; one implementing the Apriori approach, and one implementing the MAUT approach.

Key success criteria for a compound critiquing recommender are: *recommendation efficiency*, *recommendation and critique quality*, and *system usability*. Ideally, an effective system should: (1) quickly guide a user through a product-space (usually short recommendation sessions are preferred); (2) be capable of incrementally presenting the user with sugges-

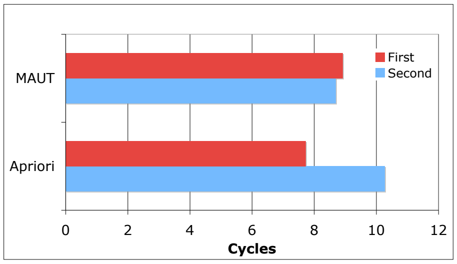


Figure 3: Average session lengths.

tions in line with their preferences through the application of compound critiques; and, (3) provide an interaction environment that is intuitive, easy to use and understand.

For the evaluation, we designed and implemented an online recommender for laptop computers, and evaluated it in accordance with the above criteria. The product database contained descriptions over 400 currently available laptops. Each laptop is described in terms of 10 features (e.g. *Price*, *Brand*, *Processor Speed*, *RAM*, etc.). The system provides a user interface that allows users to navigate through the product-space using a combination of unit and compound critiques (See Figure 2). A total of 83 users participated in the trial, evaluating both approaches. They were instructed to interact with the system to find a laptop that they would be willing to purchase and they were provided with a brief description of the recommender interface, explaining the use of unit and compound critiques. The order in which the different systems were presented was randomized. After each recommendation session users completed a usability questionnaire on on their interaction experience.

Recommendation Efficiency

For this comparison we measure the average number of recommendation cycles it takes for users to reach their target product of preference. Figure 3 illustrates the results we found here, in accordance to the order each approach was evaluated. An important point to note here is sometimes users evaluated the Apriori system first, and other times the MAUT was presented to them first to eliminate any bias as a result of learning.

Interestingly we find that user familiarization with the system and domain from their first system interaction did not lead to large efficiency improvements when using the second system. The average session-lengths for the MAUT system are relatively stable; 8.9 cycles when presented first, 8.7 cycles when second. The results from the Apriori system are slightly more variable; 7.7 cycles when first, 10.27 cycles when second. Although Apriori produces shorter sessions (by 1 cycle over MAUT), it also produces the longest (also by 1 cycle). However, there is no significant difference between the Apriori system and the MAUT system (T-Test $p = 0.97$) when we put both the first and second trial data together. Overall, both recommenders are quite efficient. From a database of over 400 laptops, both are able to recommend laptop that users are willing to buy in 10 cycles or less.

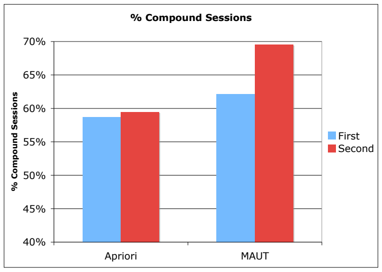


Figure 4: The proportion of sessions in which compound critiques are selected.

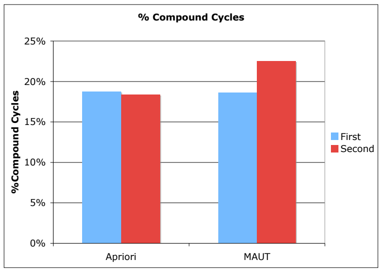


Figure 5: The proportion of cycles in which compound critiques are selected.

Application Frequency

Evaluating the quality of the compound critiques that are presented to the user involves looking at the application frequency of the generated critiques for each approach. Previous studies have shown that frequent usage of compound critiques is correlated with shorter sessions. Higher application frequencies would indicate that users find the compound critiques more useful. Focusing on those recommendation sessions where the user applied at least one compound critique, Figure 4 shows application frequency characteristics for both critique generation approaches. For the Apriori system, 58.7% (when presented first) and 59.5% (second) of sessions included at least one compound critique. The MAUT recommender fared slightly better. When presented first during the evaluation, 62% of sessions contained a compound critique. When second, this rose to almost 70%. It is worth highlighting that application frequency is higher when the recommenders were presented second indicating that perhaps it took the users a number of cycles before they appreciated the effectiveness of the compound critiques.

Figure 5 shows the proportion of cycles for which compound critique was chosen over unit critiques for both systems. Users of the Apriori system chose compound critiques between 18.7% and 18.8% of the time. When presented with the MAUT system first, users chose compound critiques 18.3% of the time. When presented second, this rises to 22.5%. These application frequency results are consistent with previous real-user trials. The differences between the systems are not significant ($p = 0.43$).

User Satisfaction

Following the evaluation we presented users with a post-study questionnaire in order to gauge their level of satisfaction with the system. For each of 11 statements (see Table 1).

Table 1: Evaluation Questionnaire

ID	Question Description
S1	I found the compound critiques easy to understand.
S2	I didn't like this recommender, and I would never use it again.
S3	I did not find the compound critiques informative.
S4	I found the unit-critiques better at searching for laptops.
S5	Overall, it required too much effort to find my desired laptop.
S6	The compound critiques were relevant to my preferences.
S7	I am not satisfied with the laptop I settled on.
S8	I would buy the selected laptop, given the opportunity.
S9	I found it easy to find my desired laptop.
S10	I would use this recommender in the future to buy other products.
S11	I did not find the compound critiques useful when searching for laptops.

The agreement level ranked from -2 to 2, where -2 is strongly disagree, and 2 is strongly agree. We were careful to provide a balanced coverage of both positive and negative statements so that the answers are not biased by the expression style. A summary of the responses we collected is shown in Figure 6.

From the results, both systems received positive feedback from users in terms of their ease of understanding, usability and interfacing characteristics. Users were satisfied with the recommendation results retrieved by both approaches (see $S2$ and $S7$) and found the compound critiques efficient (see $S5$). The results generally show that compound critiquing is a promising approach for providing recommendation information to users and most indicated that they would be willing to use the system to buy laptops (see $S2$ and $S10$).

Some interesting results can be found if we compare the average ranking level of both systems. Participants indicated on average a higher level of easy understanding in MAUT approach (see $S1$, 1.18 vs. 0.86, $p = 0.006$), which shows that compound critiques provided by the MAUT approach are easier to understand. Also, on average users ranked the MAUT approach more informative (see $S3$, -0.59 vs. -0.18, $p = 0.009$). Moreover, users are more likely to agree with the statement that the unit-critiques are better at searching for laptops with Apriori approach than the MAUT approach (see $S4$, 0.82 vs. 0.41, $p = 0.01$). Responses to our other questionnaire statements showed no significant difference between the two critique generation approaches.

SUMMARY & FUTURE WORK

In this paper two research groups from different institutions have come together to carry out a comparison of two product recommender systems that differ the way they generate and use compound critiques. Our findings with respect to their recommendation efficiency, recommendation and critique quality and interfacing satisfaction, show that both approaches are effective at navigating users to suitable products. Future work efforts will now focus on investigating ways of marrying ideas from both approaches to further improve on the results reported here. This will involve a thorough analysis of the interaction logs that have been collected as a result of this first trial.

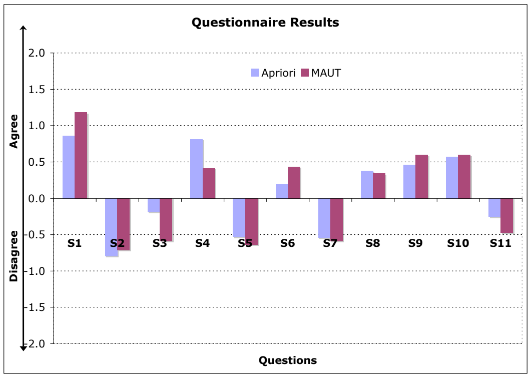


Figure 6: The results from questionnaires.

ACKNOWLEDGEMENTS

This material is based on works supported by Science Foundation Ireland under Grant No. 03/IN.3/I361 and by Swiss National Science Foundation under grant 200020-111888. We are grateful to the participants of the user studies.

REFERENCES

1. R. Agrawal, H. Mannila, R. Srikant, H. Toivonen, and A. Inkeri Verkamo. Fast discovery of association rules. In *Advances in Knowledge Discovery and Data Mining*, pages 307–328. AAAI/MIT Press, 1996.

2. R. Burke, K.J. Hammond, and B.C. Young. The findme approach to assisted browsing. *IEEE Expert*, 12(4):32–40, 1997.

3. R.L. Keeney and H. Raiffa. *Decisions with Multiple Objectives: Preferences and Value Tradeoffs.* John Wiley and Sons, New York, 1976.

4. K. McCarthy, J. Reilly, L. McGinty, and B. Smyth. Experiments in dynamic critiquing. In *Proceedings of the 10th International Conference on Intelligent User Interfaces (IUI '05)*, pages 175–182. ACM Press, 2005. San Diego, CA, USA.

5. J. Reilly, K. McCarthy, L. McGinty, and B. Smyth. Dynamic critiquing. In *Proceedings of the 7th European Conference on Case-Based Reasoning (ECCBR-04)*, pages 763–777. Springer, 2004. Madrid, Spain.

6. J. Reilly, K. McCarthy, L. McGinty, and B. Smyth. Incremental critiquing. In *Research and Development in Intelligent Systems XXI. Proceedings of AI-2004*, pages 101–114. Springer, 2004. Cambridge, UK.

7. J. Zhang and P. Pu. A comparative study of compound critique generation in conversational recommender systems. In *Proceedings of the 4th International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH2006)*, pages 234–243. Springer, 2006.