Bridging the Gap: Using IR Models for Evaluating Exploratory Search Interfaces

Max L. Wilson, m.c. schraefel
School of Electronics and Computer Science
University of Southampton, UK
[mlw05r, mc]@ecs.soton.ac.uk

ABSTRACT
Exploratory Search Interfaces are being designed to support increasingly high-level search activities, such as comparison and aggregation. This position paper suggests that the history of research into user search behaviour may provide grounds for developing a combined model approach that evaluates features designed to support such exploratory search activities.

Author Keywords
HCI, Searching, Search, Exploratory, Exploratory Search, Theory, Evaluation, Comparison

ACM Classification Keywords
H.5.2 User Interfaces, H.1.2 User/Machine Systems, H.3.3 Information Search and Retrieval

INTRODUCTION
Keyword searching has become the standard model of search, through the popularity of web giants like Google. Yet research has become directed at richer modes of search, known as Exploratory Search [9]. Naturally, some recent Human Computer Interaction (HCI) research has been investigating the design of Exploratory Search Interfaces (ESIs) for supporting users who have less clear or more complex needs.

There are many ways to design support for more complex modes of search and it would be useful to frame ESI development in an approach that HCI designers can apply to evaluate their work. A first approach may be to see if Information Retrieval (IR) research into user search behaviour, dating back to the 1980s [1, 5], can be used to model the success of any potential designs.

In this position paper, we propose an approach that combines two established IR models for evaluating ESIs. We show briefly how this approach might be applied to example ESIs, in this case three Faceted Browsers, which identified strengths and weakness in feature design for the support for users in different conditions. In the conclusion, we touch on the next steps for expanding this approach.

RELATED WORK
Exploratory search has been defined as more complex IR activities that may be required by users who do not have clearly defined goals, have changing complex needs or may be using a system that is poorly indexed [9]. Marchionini suggested some of these possible activities, including: Aggregation, Comparison and Evaluation [4]. In these conditions, simple keyword search may not support users effectively and convoluted user coping strategies have been recorded that involve iterative tentative guessing of keywords [6]. Clearly, what these users require is an alternative method of search that may involve more browsing and exploring activities to achieve their goals. This has been the seed for developing ESIs that support many richer modes of search.

Faceted browsers are an example class of ESIs, which present meta-data attributes as a series of selectable categorised options. Through modelling the domain of information through a faceted structure, direct manipulation can be used to construct queries. Thus, when a user is not clear on appropriate terminology or the meta-data is unpredictable, they do not have to estimate search terms, but can make selections to build their query. Through this extra support, faceted browsers can be considered a type of ESI.

There are many more features that can support Exploratory Search. For example, mSpace [8] is a faceted browser that also includes: a collection space, a multimedia preview cue and a persistent panel for contextual information. Further, each of these main features are designed to be persistently in view and available for use. As more features are developed and added to software, there must be rationale for how and when to expose features to support users effectively. This can not be done without evaluating how easily these features can be used and when they might be appropriate. Models of user search behaviour may be able to provide this information.

Traditionally, there are two overview styles of model in IR research: Holistic System models and Interactive models. Saracevic's stratified model [7] is an example of a holistic model that describes the different layers of an IR system: both computer and human. While both contain cascading and interacting levels, here we are concerned with analysing *user* search behaviour. The user has cascading levels including, from the top: Situational, Affective, Cognitive and Query Generation. That is, mental query generation is affected by their situational tasks, the affective intentions and their existing cognitive knowledge. It should be important to consider each of these aspects of a user when defining complex needs and subsequent requirements for a system.

Belkin *et al.* developed one of the more popular interactive models of IR, defining a series of typical episodes and scripts for the interaction between users and IR systems [2]. Included in this research, however, was an initial model for situational and cognitive user needs. They identified four dimensions, which together produce 16 unique Information-Seeking Strategy (ISS) conditions. *Method* describes whether a user is either searching for an information object, or scanning a set of information objects. This is easily differentiated by finding a specific paper in order to get its reference details, or by searching for an possible paper, which may not exist, that can be used to support a point. *Goal* describes whether a user is learning about something or selecting something. Using the bibliographic example differentiates these as researching a topic, or finding a reference. *Mode* is between recognising and specifying something. One might remember that there was a useful publication at CHI2005 and so is trying to identify it in the proceedings, or may have known the author, title and year and has typed them into the ACM Portal[1]. *Resource* is between wanting information items or meta data about an information item. Usually, with a bibliographic repository users are trying to find specific papers, but it is possible that the user is trying to find out first what workshops existed in a conference so that they can better define a search query at a later point in time.

In 1990, Bates described a four-level hierarchy of search activities: Move, Tactic, Stratagem and Strategy [1]. The first of these is a single action performed by the user, either physically or mentally: mental actions may be deciding or reading. A tactic is a combination of moves and there are endless combinations of moves that can be used to support a tactic, which depends on system implementations. She defines 32 specific information search tactics that a search system should support. Stratagems are a larger combination of both individual moves and tactics: some examples include performing a citation search or following a footnote. Marchionini noted a series of exploratory search activities, which could be considered as Stratagems [4]. Exploratory search activities could be considered complex combinations of tactics and moves, whereas a simple lookup could be a simple set of tactics and moves. Strategies are again higher and involve a combination of moves, tactics and stratagems: this might be finding relevant work for a paper and depends heavily on what the user is currently working on.

PROPOSED MODEL

Design

When considering users of a system, a user's strategy may have led the user to the system, as it usually represents a certain type of resource, such as a journal archive or a product collection. These strategies relate quite well to Saracevic's situational tasks. Once using a particular resource, the user may wish to employ a set of stratagems to achieve their goal, and modelling these in full should relate directly to Saracevic's affective intentions level. Similarly, tactics, may also be considered part of Saracevic's affective intentions. Finally, moves can relate to Saracevic's query generation, as each action should contribute towards exploring the information set. The missing level of Saracevic's model is the cognitive level, defining existing knowledge, for example. This can be modelled by using Belkin's user conditions, which incorporates things such as previous knowledge. This model actually touches on a few levels of Saracevic's model, including intention.

As a result a combination of Bates' and Belkin's models has been suggested. Bates' moves are used to quantify the support for tactics by interface features. Then these tactics are applied to support Belkin's dimensions, so that the support for the sixteen conditions can be calculated.

Stage 1: Feature Identification. First, the interface features and their interactions must be identified. For example, mSpace has a set of features including: browser columns, a collection space, a preview player and an information panel. The features of each design should be incorporated.

Stage 2: Measuring Support for Tactics. Each interface feature is addressed one at a time, for each design. For the current feature of the current design, the moves required to support each tactic are counted. This produces a series of tables, one for each design, where tactics are listed across the top and the interface features down the side. The count of moves is noted in the appropriate cross section between feature and tactic. No support by a feature for a tactic counts as 0. Four moves for the user to use the feature in support of a tactic counts counts as 4. Repeat and Optional moves are ignored. For example, selecting multiple items involves choosing and selecting 2+ items, selecting 3+ is considered a repeat move of selecting 2 items. Optional moves include scrolling: a desired item may be the first or last item. The optimum situation is that it is one of the items that is visible without scrolling.

Stage 3: Summarising Metrics. As no support is represented by zero, support in a single move is represented by 1 and support in ten moves by 10, all values above 0 must be inverted. Thus a feature that supports a tactic well approaches the value of 1 and a poor support approaches 0. These inverted metrics can then be summed by feature and by tactic. This calculates the support provided by a feature for all tactics and the support provided for a tactic across all features, respectively.

Stage 4: Feature Strength Analysis. A graph can be produced including the summed values for each feature in each design. An example can be seen in the following section. Strong features will score produce tall bars, and a quick comparison of user effort can indicate a strong feature design.

Stage 5: Tactic Support Analysis. A graph can be produced including the summed values for each tactic in each design. Again, tall bars indicate strong support for a tactic. This comparison may identify tactics which may require improved support through redesign.

[1]http://portal.acm.org

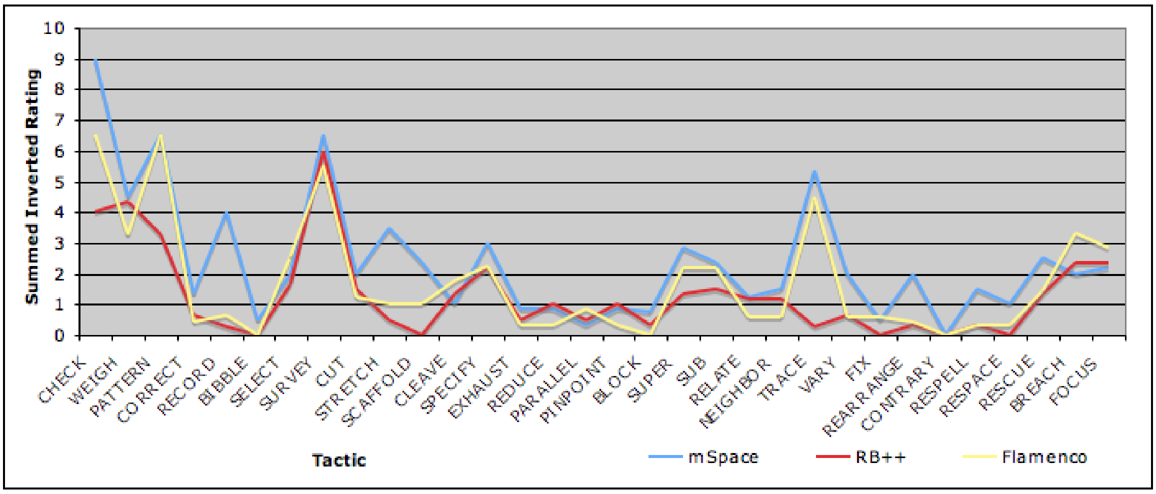

Figure 1. Graph Showing the Summed Inverted Metrics of each Tactic for the Three Browsers.

Stage 6: User Conditions Analysis. Each tactic supports particular ends of Belkin's dimensions of user conditions. CHECK, a tactic for users checking their decisions so far, supports users who are trying to Learn as their *Goal*. The support for a tactic by a design is added to the total support for a dimension. Then for each of the sixteen conditions, the sum of the total support values are calculated. This value for each condition can be graphed showing the difference in support for different user conditions.

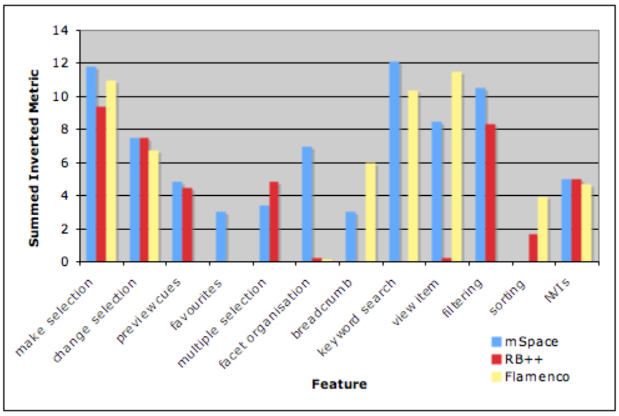

Figure 2. Graph Showing the Summed Inverted Metrics of each Feature for the Three Browsers. Taller bars are better.

Model Applied

In a recent application of the model, three faceted browsers were compared: mSpace[2], Flamenco[3] and RB++[4]. It is clear from Figure 1 that there are few tactics that are not best supported by the mSpace browser. This is explained, in Figure 2 by the number of strong contending features: there are comparably high mSpace bars for almost every feature. This is arguably representative of the focus+context design, which aims to present as many options and features to the user as possible and at all times. In line with the second observation, however, there are clearly some features of each interface that have stronger implementations of the three browsers. For example, multiple selection is easiest in RB++, yet keyword search is missing from RB++ and the implementation is strongest in mSpace. One feature missing from the mSpace implementation is the ability to sort items. The strongest implementation of this is the ability to group the results by any facet, as seen in Flamenco.

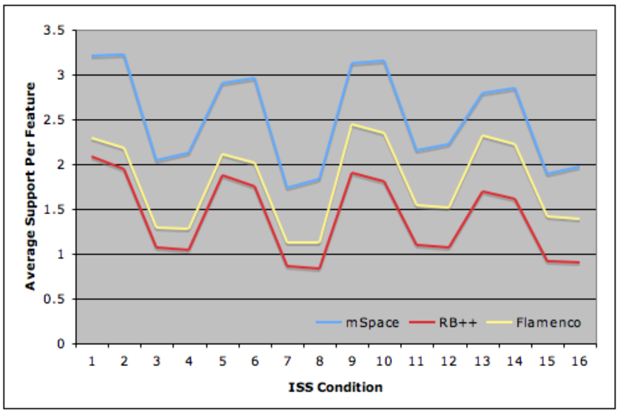

Figure 3. Graph Showing the Normalised Support for each ISS Condition by Faceted Browser

The pattern that is seen almost identically for each interface in Figure 3 is indicative of the mapping between Bates' tactics and the pattern of ISS conditions defined by Belkin *et al.* Predictably, as was shown in Figure 1, there are three distinct lines, showing that mSpace provides the widest support for search. This height difference does not show us new information. Instead what should be drawn from the graph is hidden within this pattern and shown in the differences in peaks and troughs for each interface condition. Quite clearly the graphs rise and fall in alternating pairs. This represents the alternation between recognise and specify (*Mode*) and is perhaps a predictable outcome for faceted

[2]http://mspace.fm
[3]http://flamenco.berkeley.edu/
[4]http://idl.ils.unc.edu/rave/

browsers. By including more lessons learnt from the IR work on keyword search, such as relevance feedback, we might see a balance between these two conditions. Within each of these alternating pairs, the mSpace line marginally increases where the others fall. This indicates an increased support for meta-information (*Resource*). Considering individual browser lines, while RB++ and Flamenco follow a similar pattern for the first 8 ISS conditions, Flamenco notably improves this gap in the final 8 conditions. These two halves are made unique by the *Method* dimension and indicates that Flamenco provides better support for search, which is defined by having a known target object to exist: this might be knowing that an academic paper exists and just trying to find it. This significant increase, also sharper than mSpace, may be present due to the better support for advancing selections and the lower support for changing selections.

The final pattern we draw from Figure 3 is shown every four conditions and is controlled by Belkin's *Goal* dimension. The Learn aspect of this dimension is shown by height differences between ISS1-4 and ISS5-8, and again between ISS9-12 and ISS13-16. This is characterised by the ability to see options in faceted browsers. The persistence of these options shown throughout to the user of mSpace is highlighted by the exaggerated difference in the first and third troughs compared to the second and fourth.

OPEN RESEARCH QUESTIONS

There are a few areas where this approach could be extended. First, the definition of a single move is still open to some interpretation. In the application above, choosing mentally and selecting physically are separated as different moves. Cognitive models exist that could be used to define moves in a clear manner. Further, cognitive load is an increasingly important problem for rich interfaces, and the effect of adding more features should also be modelled; an interface with 100 features may support many tactics but may not be easy to use. This is already modelled to some extent, as it may take more moves to access a feature if you have to select and open it before it can be used.

Of the four user levels of Saracevic's stratified model, there are still gaps for improvement. Belkin is one model of user intentions, but more exist. The same team later developed user and search conditions into much greater detail [3]. Further, Saracevic's affective intentions can be better modelled by trying to define Stratagems like Marchionini's exploratory search activities. These activities in particular are largely undefined, but where definition is missing, a model may be available or developed to extend this research. Also, this research is currently trying to model the user levels defined by Saracevic in 1997. There may be more recent or more specific research that models the user in more detail.

Finally, the current model is only useful when comparing designs and implementations. Ideally the model can identify a strong design or a weak design independently, so that it can be applied where there is no benchmark for comparison.

CONCLUSION

In this paper, we have described an approach for evaluating Exploratory Search Interfaces for strengths and weaknesses in terms of feature design for the support of users in different search conditions. This approach combines two established models of Information Retrieval to quantify a comparison of three Faceted Browsers, as example Exploratory Search Interfaces, and identified some clear areas for improvement on each. The approach can be applied to existing implementations, as above, or to new designs for exploratory search interfaces. Designs can be measured in their support for search tactics and strengthened before user studies begin.

These are early steps to a potentially promising approach for blending IR frameworks with HCI design. Based on these initial studies, we are investigating several refinements of the approach. In particular we are considering strategies to correlate the predictive findings presented here against experimental studies to validate and refine the approach proposed. We anticipate that with such validation, this hybrid IR/HCI approach will be a useful design tool for the development of exploratory search interfaces.

REFERENCES

1. BATES, M. J. Where should the person stop and the information search interface start? *Inf. Process. Manage. 26*, 5 (1990), 575–591.
2. BELKIN, N. J., MARCHETTI, P. G., AND COOL, C. Braque: design of an interface to support user interaction in information retrieval. *Inf. Process. Manage. 29*, 3 (1993), 325–344.
3. COOL, C., AND BELKIN, N. J. A classification of interactions with information. In *Proc. CoLIS 4* (2002), pp. 1–15.
4. MARCHIONINI, G. Exploratory search: from finding to understanding. *Commun. ACM 49*, 4 (2006), 41–46.
5. PEJTERSEN, A. M. A library system for information retrieval based on a cognitive task analysis and supported by an icon-based interface. In *Proc. ACM SIGIR* (New York, NY, USA, 1989), ACM Press, pp. 40–47.
6. PIROLLI, P., AND CARD, S. Information foraging in information access environments. In *Proc. ACM SIGCHI* (New York, NY, USA, 1995), ACM Press/Addison-Wesley Publishing Co., pp. 51–58.
7. SARACEVIC, T. The stratified mode of information retrieval interactino: Extension and applications. In *Proc. Info. Science* (1997), vol. 34, pp. 313–327.
8. SCHRAEFEL, M. C., WILSON, M. L., RUSSELL, A., AND SMITH, D. A. mspace: improving information access to multimedia domains with multimodal exploratory search. *Commun. ACM 49*, 4 (2006), 47–49.
9. WHITE, R. W., KULES, B., DRUCKER, S. M., AND M.C. SCHRAEFEL. Introduction. *Commun. ACM 49*, 4 (2006), 36–39.