Supporting Exploratory Search with a Visual User-Driven Approach

CECILIA DI SCIASCIO and VEDRAN SABOL, Know-Center GmbH
EDUARDO VEAS, National University of Cuyo

Whenever users engage in gathering and organizing new information, searching and browsing activities emerge at the core of the exploration process. As the process unfolds and new knowledge is acquired, interest drifts occur inevitably and need to be accounted for. Despite the advances in retrieval and recommender algorithms, real-world interfaces have remained largely unchanged: results are delivered in a relevance-ranked list. However, it quickly becomes cumbersome to reorganize resources along new interests, as any new search brings new results. We introduce an interactive user-driven tool that aims at supporting users in understanding, refining, and reorganizing documents on the fly as information needs evolve. Decisions regarding visual and interactive design aspects are tightly grounded on a conceptual model for exploratory search. In other words, the different views in the user interface address stages of awareness, exploration, and explanation unfolding along the discovery process, supported by a set of text-mining methods. A formal evaluation showed that gathering items relevant to a particular topic of interest with our tool incurs in a lower cognitive load compared to a traditional ranked list. A second study reports on usage patterns and usability of the various interaction techniques within a free, unsupervised setting.

CCS Concepts: • **Information systems → Search interfaces**; • **Human-centered computing → Human computer interaction (HCI)**; **Interactive systems and tools**; **Empirical studies in HCI**; *Visual analytics*;

Additional Key Words and Phrases: Exploratory search, advanced search interface, textual document ranking

ACM Reference format:
Cecilia di Sciascio, Vedran Sabol, and Eduardo Veas. 2017. Supporting Exploratory Search with a Visual User-Driven Approach. *ACM Trans. Interact. Intell. Syst.* 7, 4, Article 18 (December 2017), 35 pages.
https://doi.org/10.1145/3009976

1 INTRODUCTION

With the advent of electronic archival, the skill to find and organize the right information has become paramount, as searching for and collecting information occupy a large portion of our daily productive time. Information seeking is a widely studied phenomenon [19, 54, 57]. We consider the problem beyond *lookup* search (where the information need of the user is assumed static [19]),

The reviewing of this article was managed by special issue associate editors Massimo Zancanaro and Cristina Conati.
This work is funded by the European Union's Seventh Framework Programme (FP7/2007–2013), project EEXCESS under Grant No. 600601. The Know-Center GmbH is funded by the Austrian COMET Program – managed by the Austrian Research Promotion Agency (FFG).

Authors' addresses: C. D. Sciascio, V. Sabol, and E. Veas, Inffeldgasse 13, 8010 Graz, Austria; emails: {cdisciascio, vsabol, eveas}@know-center.at.

https://doi.org/10.1145/3009976

in a context where finding and organizing information is a dynamic process with sudden changes in information needs occurring as the user learns [19, 32].

Advanced search engines and recommender systems (RSs) have grown as the preferred solution for contextualized search by narrowing down the number of entries that need to be explored at a time. Traditional information retrieval (IR) systems strongly depend on precise user-generated queries that should be iteratively reformulated to express evolving information needs. Arguably, a user constructs knowledge throughout this iterative process, but IR interfaces hardly support this process.

In addition, formulating queries has proven to be more complicated for humans than plainly recognizing information in a visual manner [19], which is why the combination of IR with machine learning and HCI techniques has led to a shift toward—mostly web based—browsing search strategies that rely on on-the-fly selections, navigation, and trial and error [32].

In the case of RSs, trust issues may hinder user engagement during exploration. As Swearingen et al. [58] pointed out in their seminal work, the RS has to persuade the user to try the recommended items. Thus, not only does the recommendation algorithm have to fetch items effectively, but also the user interfaces must deliver recommendations in a way that they can be compared and explained [47]. User-centric interfaces aim to boost trust in recommended items by explaining how the system works and makes decisions (*transparency*) [21, 59] or by allowing users to tell the system when it is wrong (*scrutability*) [26]. Hence, to warrant increased user involvement, the RS has to justify recommendations and let the user personalize their generation. Explanatory interfaces help establish a connection between the workings of the RS, needs, and recommendations.

Exploratory search is actually part of a discovery process in which the user often becomes familiar with new terminology in order to filter out irrelevant content and spot potentially interesting items. Such process requires careful inspection of at least a few titles and abstracts, when not full documents, before becoming familiar with the underlying topic.

Along the way, users build knowledge as cognitive constructs of topics related by resources. In Hearst terms, “users learn about the topic as they scan retrieval results and term suggestions, and formulate new sub-questions as previously posed sub-questions are answered” [19]. For example, after inspecting a few documents related to “robots,” subtopics like “human-robot interaction” or “virtual environments” could attract the user’s attention.

Our work tackles exploration and production tasks that involve gathering and organizing large collections in preparatory steps, for example, for writing or preparing a lecture or a presentation. We investigate the discovery process and associated strategies by deploying an interface emphasizing transparency, controllability, and predictability as key features to support (1) exploration of textual document recommendations and (2) refinement of evolving information needs. *uRank* is a visual analytics approach that automatically generates an interactive keyword-based overview of the document collection. It provides visual hints that allow the user to preview the effect of selecting a keyword (in terms of bearing documents) and discover frequently co-occurring keywords (that can be used to build key phrases). Users can refine their interests through a drag-and-drop mechanism that updates a document ranking, representing document relevance scores and query term contribution as stacked bars. We delineate the motivation behind our approach and present the results of two user studies analyzing behavior and strategies in exploratory search.

1.1 Exploratory Search

The ability to analyze and organize large collections, to draw relations between pieces of evidence, and to build knowledge are all part of an information discovery process. As a result of the discovery process, one builds a body of knowledge, a collection of resources related, for example, by topics or other aspects that is mostly a cognitive construction. Exploratory search, as proposed by Marchionini, entails *lookup*, *learn*, and *investigate*. But standard search and retrieval hide most of its mechanisms and hardly support the user in building relations and knowledge in itself [32]. It is clear that for an exploratory engine it is not enough to know just what we previously searched for; it also needs to know what we did with what we found. Was it categorized, was it good, and did it lead to another topic and another search?

Indeed, collecting information about a new topic is rarely solved with a single query [34]. It is instead a discovery process involving several queries intermingled with extensive reading of retrieved resources. Query terms are refined and reformulated as results are explored. Interesting results are collected along each step, but the connections between them exist only at a cognitive level in the user. The effort and time are mostly spent in careful reading and acquiring information from search results (often reading titles, abstracts, and text summaries) and in formulating new concepts to continue exploring. Discovery can be seen as a result of *sensemaking*—an iterative process of formulating a conceptual representation from a large volume of information [39].

Exploration can be studied as a *berry-picking* model, based on the assumption that the information need of the user continually shifts in the process of reading and learning from the information encountered [5]. Hence, the searcher's information need is satisfied by a series of selections and bits of information found along the way.

Information foraging theory models people's navigation in information structures as a reflection of evolutionary behavior learned for exploring the environment in search of food [38, 40]. In a nutshell, for animals that forage in patches, the longer they stay in a single patch, the less energy they consume (diminishing return). Much like animals, information seekers must decide whether it is more profitable to find a new patch or continue exploring the current one. Information scents are cues leading to information, for example, summarizing what can be found in an outlink [40]. For instance, when collecting scientific information, strong scent can come from references, keywords and phrases for unknown terms, and so forth. In light of the information foraging theory, information scents are called cues in visual representations. Visual representations are in turn also grounded in evolutionary cognitive models, such as saliency, a mechanism that guides the deployment of visual attention [24].

To search, a user must know his or her information need and be able to express it in a query. When browsing, users consume information of a resource and, when their information need shifts, navigate by following links. In cognitive terms, searching (querying) is demanding: it involves identifying a need, planning, executing queries, evaluating results, and refining, while browsing requires the user to identify promising links [4]. It is usually easier for a person to recognize something when looking for it than it is to figure out how to describe it. New principles such as interactive intent modeling have emerged, with a focus on deciphering user interests in order to narrow down the information space [48].

In our goal to personalize the discovery of scientific information, we built a user interface using visual attention principles to guide information scent and provide automatic mechanisms for users to explore document collections along their needs, pick interesting items, and build knowledge.

1.2 Requirements

Our solution intends to reduce the effort spent in the discovery process. In a nutshell, it aims to assist the user in quickly scanning a set of documents and in collecting those that meet his or her needs, *before* having to thoroughly read every title and/or abstract. To meet these goals, we established three major requirements as design guidelines:

RI. Recognition over recall: whenever possible, encourage the user to browse and navigate instead of requesting to reformulate queries.

RII. Sudden shifts in information needs: allow the user to reorganize the collection on the fly according to the new interests.

RIII. Transparent logic: give the user visual evidence as to why a document ranks better than another, and provide information scent to promising information; that is, the visual representation should be self-explanatory.

2 BACKGROUND

2.1 Search Result Visualization

Modern search interfaces assist user exploration in a variety of ways. For example, query expansion techniques [46] address the query formulation problem by leveraging stored related concepts to help the user extend the initial query. Recent trends lead toward a personalization of the retrieval process by modeling the underlying task [2] or user intents [13, 49].

As for document relevance explanations, tile-based visualizations like *TileBars* [18] and *HotMap* [22] became very popular because they make efficient use of space by encoding the relative frequency of query terms with shaded blocks. In the case of the former, it also conveys query term distribution within documents and relative document length. This paradigm aims to foster analytical understanding of Boolean-type queries, and hence they do not yield any rank or relevance score. All these approaches rely on the user being able to express precise information needs and do not support browsing-based discovery within the already available results.

Faceted search interfaces allow for organizing or filtering items throughout orthogonal categories, proving their usefulness for inspecting enriched multimedia catalogs [51, 64]. More recently, interfaces supporting rather natural facet-type visual filtering (e.g., geographic or temporal) have also been proposed [45]. However, faceted search relies on structured information (i.e., metadata categories), and thus it hardly supports topic-wise exploration of unstructured data.

Rankings conveying document relevance have been discouraged as opaque and underinformative [18]. However, the advantage of ranked lists is that users know where to start their search for potentially relevant documents and that they employ a familiar format of presentation. A study [53] suggests that (1) users prefer bars over numbers or the absence of graphical explanations of relevance scores, and (2) relevance scores encourage users to explore beyond the first two results. As a drawback, lists imply a sequential search through consecutive items and only a small subset is visible at a given time [55]; thus, a list alone is mostly apt for sets no larger than a few tens of documents. Focus+Context and Overview+Detail techniques [22, 44] sometimes help overcome this limitation, while alternative layouts like *RankSpiral*'s rolled list can scale up to hundreds and perhaps thousands of documents [55].

Other approaches complement ordered lists with visual metaphors [33], similarity-preserving layouts [14], or POI-based visualizations [1, 36]. Yet an unintuitive context switching is a potential problem when analyzing different aspects of the same document.

Although ranked lists are not a novelty, our approach attempts to leverage the advantages they provide (i.e., user familiarity). We augment them by adding a transparent stacked-bar-based representation for document relevance and query term contribution. *Insyder*'s bar graph [46] is an example of augmented ranked lists that displays document and keyword relevance with disjoint horizontal bars aligned to separate baselines. Although layered bar dispositions are appropriate for visualizing distribution of values in each category across items, comparison of overall quantities and the contribution of each category to the totals is better supported by stacked-bar configurations [56]. Additionally, we rely on interaction as the key to provide controllability over the ranking criteria and hence support browsing-based methods.

LineUp [15] has proven the simplicity and usefulness of stacked bars to represent multiattribute rankings. Despite targeting data of a different nature—*uRanks*'s domain is rather unstructured with no measurable attributes—the visual technique itself served as inspiration for our work.

2.2 Recommending Interfaces

In recent years, considerable efforts have been invested into leveraging the power of social RS through visual interfaces [28, 35]. As for textual content, *TalkExplorer* [60] and *SetFusion* [37] are examples of interfaces for exploration of conference talk recommendations. The former is mostly focused on depicting relationships among recommendations, users, and tags in a transparent manner, while *SetFusion* emphasizes controllability over a hybrid RS. Rankings are not transparent though, as there is no explanation as to how they were obtained. Kangasraasio et al. [25] highlighted that allowing the user to influence the RS is important, as is adding predictability features that produce an effect of causality for user actions.

With *uRank*, we intend to enhance predictability through document hint previews, allow the user to control the ranking by choosing keywords as parameters, and support understanding by means of a transparent graphic representation for scores.

2.3 Topic Analysis

In addition to methods for visualizing item relevance or distribution of query terms along search results, we consider it appropriate to also include in this section another group of methods that support exploration by providing a topical overview of a document set.

Tag clouds have been proposed for browsing document collections [52]. Besides providing a topical overview, such representations are used for keyword-based filtering but do not provide possibilities to influence a ranking.

Clustering approaches like Scatter/Gather [10] are able to handle very large pools by building hierarchical structures for top-down exploration. IN-SPIRE [29] and InfoSky [3] provide cluster browsing interfaces based on spatial metaphors: a landscape and the outer space, respectively. Hierarchical cluster exploration is not a trivial task; therefore, it is rarely adopted by real-world systems.

Another alternative is topic models, which perform a generative approach (e.g., latent Dirichlet allocation) to capture themes inherent to a document collection [6]. Typical UIs present a topic overview of a collection and allow for further exploration at multiple levels via zooming with keyword search [12] or by navigating a network of interconnected documents, as in *TopicNets* [16]. Topic models owe their flexibility to the fact that they do not correspond to any predefined taxonomy. The model generation process does not infer any semantic information; instead, it discovers patterns based on term co-occurrence. This flexibility could also turn into a weakness, as the topics generated are often not interesting or relevant to users [23]. Moreover, topic models are costly to compute and the exploration and discovery process only works on a pre-existing collection. Although it is possible to interactively change a topic model by joining or splitting topics, these methods aim at improving the model rather than supporting exploratory search [23]. We seek a solution that is not only flexible but also personalizable. Users should be able to construct their own topics as their interests evolve.

3 THE URANK APPROACH

uRank is an interactive user-driven approach that combines lightweight text analytics and an augmented ranked list to assist in exploratory search of textual documents. It aims at supporting the sense-making process by adapting to frequent shifts in user information needs and maximizing the principle of recognition over recall [19].

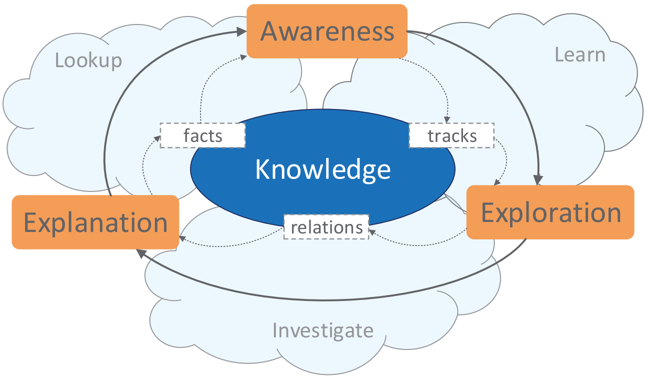


Fig. 1. Conceptual cycle of exploration and discovery. A continuous process entailing awareness, exploration, and explanation bridging Marchionini's exploratory search model [32]. As a user consumes information, he or she becomes aware of tracks to other pieces of information. Tracks add up until their information value is greater than that of the current source and the user turns to explore them.

3.1 Conceptual Cycle of Exploratory Search

Our approach focuses on building knowledge sustained in a cognitive model of exploratory search. The abstract model in Figure 1 describes Marchionini's model for exploratory search with its connecting parts: *lookup, investigate, learn* overlaid with the principles of awareness, exploration, and explanation interlocked in a continuous discovery process. When the user becomes *aware* that he or she needs more evidence, he or she chooses a path and starts a focused *exploration*. While exploring, the user follows different tracks, discovers relationships, and collects facts, in order to start building hypotheses. As hypotheses mature, the user turns to test them, seeking *explanation* in the facts collected. Each of these stages contributes knowledge, as tracks that the user becomes *aware* of, as relationships elicited through *exploration*, and as facts acquired via *explanation*. In terms of information foraging [40], awareness is guided by information scent, which sustains the choice of *tracks* to follow and patches to explore. In our case, patches are documents or subcollections thereof. Foraging is the process of building one's own topic structure along with perceived relationships in the data collection.

Figure 2 depicts the workflow between automatic and interactive mechanisms in *uRank*. Combining these mechanisms enables users to explore a document collection and refine information needs in terms of topic keywords. The workflow is summarized as follows:

(1) uRank receives a set of textual document surrogates, that is, titles and abstracts. The web-based implementation is currently fed by an RS connected to several data sources [27].
(2) The keyword extraction module analyzes all titles and abstracts and returns: (*i*) a list of weighted representative terms for each document and (*ii*) a set of keywords that describe the whole collection.
(3) The UI displays a list of documents along with the extracted collection keywords.
(4) The user explores the documents and keywords. During this process, the user can discover possible key phrases or relations between documents and keywords at a glance.
(5) When the user finds interesting terms, they can be interactively selected either individually or as group via drag and drop.
(6) The document list is re-sorted according to the specified keywords and augmented with colored stacked bars denoting document scores.

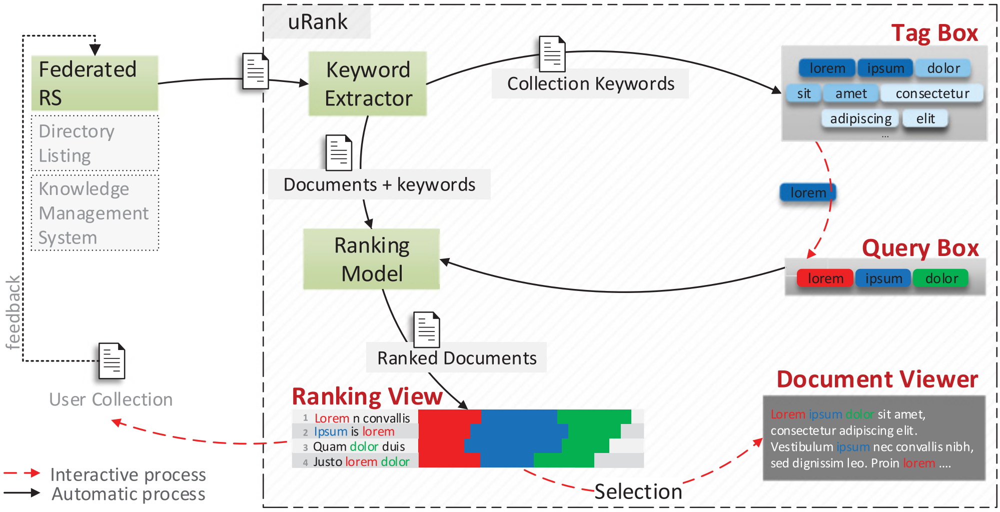


Fig. 2. *uRank*'s workflow showing automatic (black arrows) and interactive mechanisms (red arrows).

(7) The user can select a single document to access more detailed information.
(8) Once the user finds a document that suits his or her search interest, he or she can add it to his or her own collection.

User-driven actions (4, 5, 7, and 8) align to the exploration and discovery cycle in Figure 1 and strongly depend on the user's search strategy. Thus, they are likely to unfold in an iterative and interchangeable manner.

3.2 The User Interface: Interactions and Visual Design

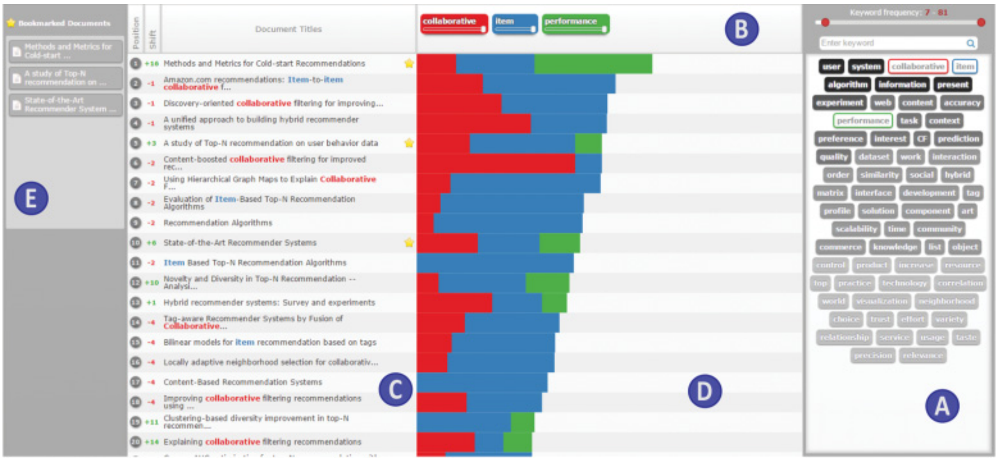
uRank's UI is arranged in a multiview fashion that displays different levels of abstraction of a document collection. The spatially separated disposition of the views entails a semantic Overview+Detail scheme [9] that aims at supporting the different stages in our information discovery cycle:

Collection overview (awareness). The *Tag Box* (Figure 3(A)) summarizes the entire collection through augmented keyword tags and provides additional hints about relationships within the data.

Documents overview (exploration). The *Document List* shows titles along with ranking information, and the *Ranking View* displays stacked bar charts depicting document relevance scores (Figures 3(C) and 3(D), respectively). Together they represent minimal document views. The list and ranking visualization are updated as the user manipulates keyword tags in the *Query Box* (Figure 3(B)).

Document detailed view (explanation). For a document selected in the list, the *Document Viewer* displays the title, snippet, and available metadata. Color-augmented keywords spotlight the distribution (and proximity) of selected keywords.

We paid close attention to Baldonado's guidelines for multiview systems [61] throughout iterative stages of the layout design. At first, all views appeared juxtaposed, avoiding multiple overlapping views. In the latest version, we added the *Bookmark Overview* (Figure 3(E)) and removed the *Document Viewer* from the main view. The latter is currently shown as a modal view when the user clicks on a particular item in the list.



Position	Shift	Document Titles
1	+16	Methods and Metrics for Cold-start Recommendations
2	-1	Amazon.com recommendations: Item-to-item collaborative f...
3	-1	Discovery-oriented collaborative filtering for improving...
4	-1	A unified approach to building hybrid recommender systems
5	+3	A study of Top-N recommendation on user behavior data
6	-2	Content-boosted collaborative filtering for improved rec...
7	-2	Using Hierarchical Graph Maps to Explain Collaborative F...
8	-2	Evaluation of Item-Based Top-N Recommendation Algorithms
9	-2	Recommendation Algorithms
10	+6	State-of-the-Art Recommender Systems
11	-2	Item Based Top-N Recommendation Algorithms
12	+10	Novelty and Diversity in Top-N Recommendation -- Analysi...
13	+1	Hybrid recommender systems: Survey and experiments
14	-4	Tag-aware Recommender Systems by Fusion of Collaborative...
15	-4	Bilinear models for item recommendation based on tags
16	-4	Locally adaptive neighborhood selection for collaborativ...
17	-4	Content-Based Recommendation Systems
18	-4	Improving collaborative filtering recommendations using ...
19	+11	Clustering-based diversity improvement in top-N recommen...
20	+14	Explaining collaborative filtering recommendations

Fig. 3. *uRank* user interface displaying documents related to *recommender systems*, with ranking updated to match the keywords "collaborative," "item," and "performance." A. The *Tag Box* presents a keyword-based summary of the document collection. B. The *Query Box* contains keywords selected by the user. The *Document List* (C) and the *Ranking View* (D) present a list with augmented document titles and stacked bars indicating relevance scores. E. The *Bookmark Overview* shows bookmarked documents.

In theory, recommender and information retrieval systems produce lists where items are already sorted by relevance with respect to certain criteria. However, it has been argued that user trust and engagement may be hindered if the UI does not provide features for reshaping the search or recommending criteria (controllability) or clear rationale as to what makes an item more relevant than another (transparency). With *uRank*, we attempt to bridge these gaps by providing a user-driven method for reorganizing documents as information needs evolve, along with a suitable visual encoding and animated transitions that convey a transparent logic for document relevance.

The remainder of this section digs deeper into visual and interactive aspects of each component of the UI dedicated to support the different steps in the discovery cycle, namely: the *Tag Box* for *awareness* (Section 3.2.1); the *Document List*, *Ranking View*, and *Query Box* for *exploration* (Section 3.2.2); and the *Document Viewer* for *explanation* (Section 3.2.3).

3.2.1 Awareness: Topic Overview and Information Tracks. *uRank* automatically extracts keywords from the recommended documents with a twofold purpose: provide (1)an overview of the collection and (2) manipulable visual elements that allow the user to express information needs. The *Tag Box* (Figure 3(A)) is the component dedicated to raise awareness and comply with **RI**.

Topic Overview. The *Tag Box* represents a summary of the textual documents in terms of keyword tags. Summarizing the collection in a few representative terms allows the user to scan the recommendations and grasp the general topic and related terms at a glance. This is particularly important in the context of collections brought by RS, where the user is normally not directly generating the queries.

Keyword tags are arranged in a bag-of-words fashion, encoding relative frequencies through position and intensity. The ordering conveys descending document frequency (DF), while five levels of gray shading allow for visually grouping keywords with similar frequencies. Redundant frequency coding is intentional and aims at maximizing distinctiveness among items in the keyword set [63].

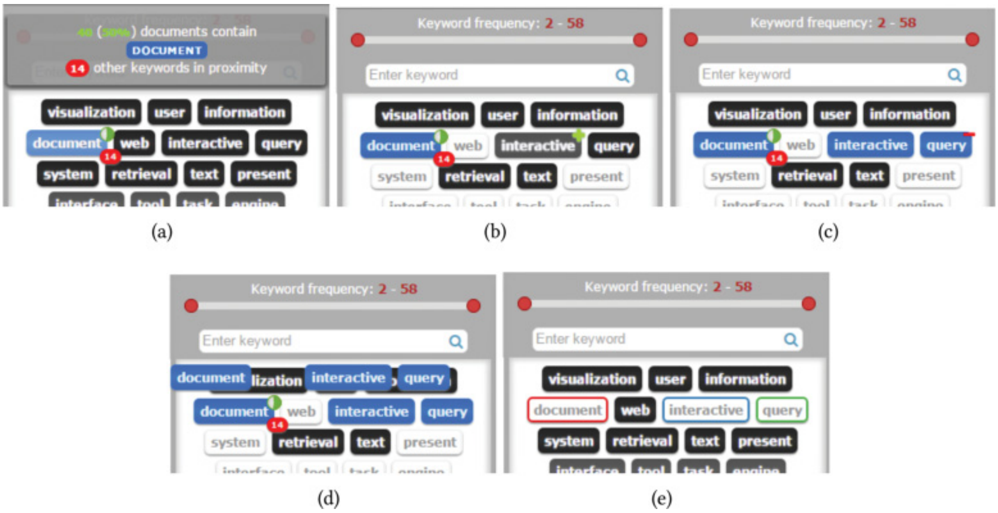


Fig. 4. *Tag Box*. (a) Keyword hints and tooltip become visible on hover. (b) Clicking on a tag locks the view and frequently co-occurring keywords are highlighted. (c) Clicking on additional tags creates a multiple selection. (d) A group of tags being dragged with the cursor. (e) Dropped tags are cloned in the *Tag Box* and highlighted with the appropriate border stroke.

Information Tracks. A bag-of-words arrangement is able to convey an outline of keywords covered in a topic and their relative frequencies. However, this representation alone is insufficient to supply further details about how a keyword relates to other keywords or documents.

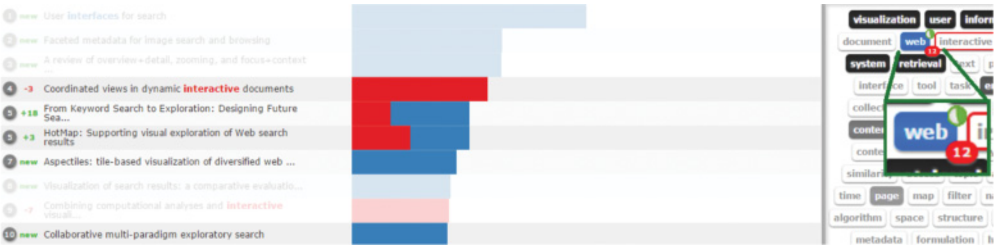
Information scent cues are incorporated by augmenting tags with two compact visual hints. The *co-occurrence hint* appears as a red circle showing the number of frequently co-occurring keywords across all documents, whereas the *document hint* consists of a pie chart that conveys the proportion of documents in which the keyword is contained. Both hints become visible on mouse hover, along with a tooltip at the top of the *Tag Box* that provides a more in-detail explanation of the hints' meaning (Figure 4(a)). In a nutshell, the purpose of these hints is to notify the user that further inquiries within the same "patch" are possible. Then, clicking on a particular tag provides the following additional tracks:

(1) Unrelated documents are dimmed in the *Document List* and *Ranking View*, so that documents containing the keyword remain in focus. This feature allows for predicting the effect of selecting a keyword (Figure 5).
(2) Co-occurring terms are brought into focus by dimming unrelated tags in the background (Figure 4(b)), in order to support the user in discovering possible key phrases within the collection.

3.2.2 *Exploration: Ranking Documents on the Fly*.

During exploration, the information needs of the user frequently change and documents are collected under different assumptions. *uRank* encourages the user to reshape the inquiries and reorganize contents on the fly, as stated in **RII**.

Document titles are initially listed following the order in which they were supplied. Changes in the document ranking visualization originate from three types of keyword tag manipulations in the *Query Box* (Figure 3(B)): addition, weight change, and deletion. Throughout this article, we often refer to tag manipulations as control features, denoting that these are the means for the user to steer the document ranking.



1 new User interfaces for search
2 new Faceted metadata for image search and browsing
3 new A review of overview+detail, zooming, and focus+context ...
4 -3 Coordinated views in dynamic interactive documents
5 +18 From Keyword Search to Exploration: Designing Future Sea...
6 +3 HotMap: Supporting visual exploration of Web search results
7 new Aspectiles: tile-based visualization of diversified web ...
8 new Visualization of search results: a comparative evaluatio...
9 -7 Combining computational analyses and interactive visual...
10 new Collaborative multi-paradigm exploratory search

Fig. 5. Information tracks. The zoomed area shows document and co-occurrence hints, helping to predict the effect of selecting the hovered tag. The tag “web” has been clicked on, and thus bearing items in the list remain in focus, as well as keywords frequently appearing together.

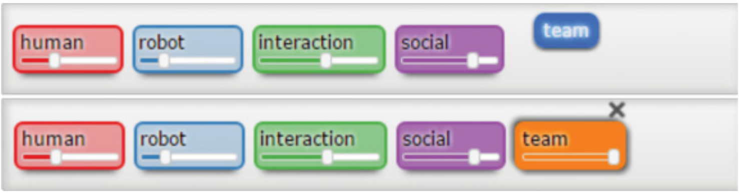


Fig. 6. (*Top*) Keyword tag before being dropped in the *Query Box*. (*Bottom*) Dropped tag extended with weight slider and “delete” button. Background colors match a categorical scale and weights have been tuned.

Tag Addition. Keyword tags in the *Tag Box* can be manually unpinned, dragged with the mouse pointer, and dropped into the *Query Box*. Tags can be dropped either one by one (Figure 6) or as a group. After clicking on a tag of interest, the *Tag Box* appears “locked” so that the frequently co-occurring tags remain highlighted. The user then can click on additional tags (Figure 4(c)) and drag them all together (Figure 4(d)). The *Tag Box* is “unlocked” once the tags are dropped into the *Query Box*. The intention of incorporating multiple drag-and-drop is to assist users in interactively creating their own key phrases. Remembering the sequence of tags that form a potentially interesting key phrase entails higher cognitive effort if the user can only drag tags individually and the sequence vanishes from the bare eye every time the view in the *Tag Box* is unlocked.

Dropped tags are re-rendered by adding a weight slider, a delete button on the right-upper corner (visible on hover), and a specific background color from a categorical palette (Color Brewer’s *9-class Set 1* qualitative palette [17]). Since keyword tags represent category labels (i.e., abstractions of entities into groups), the use of a categorical color scheme allows for clearly distinguishing tags from one another.

Weight Change. Tag sliders allow the user to adjust the weight of a keyword in document scores. Figure 6 shows tag backgrounds with different levels of intensity after the sliders have been tuned.

Tag Deletion. Tags can be removed from the *Query Box* by clicking on the “delete” icon. The user also has the alternative to clear the *Query Box* and restore the document list to its original state. In any case, animation is used to shift tags to their original positions in the *Tag Box* at a perceivable pace.

3.2.3 Explanations: Ranking Visualization and Details on Demand. Several HRI (Human-Recommender Interaction) studies describe transparency as a desirable characteristic for recommending interfaces because they can boost acceptance and trust of recommended items [21, 59]. This concept is often associated with some sort of explanation that somehow enlightens the user about the decisions made by the RS.

Here we seek to foster transparency, but from an exploratory search perspective. Instead of providing textual proof that a certain item is worth “acquiring” (and consequently ending exploration at this point), we define an interface as transparent if its graphical representations allow the user to (1) quickly compare documents in terms of their contents and figure out which ones are worth exploring in the current “foraging patch” and (2) readily validate their hypothesis as to whether a document is actually relevant or not. Thus, we rely on a suitable visual encoding and animated transitions to comply with **RIII**. Visual encoding supports preattentive processing by leveraging the capacity of human vision to absorb great amounts of information at a glance. In turn, interactions enable users to directly or indirectly manipulate the data through the view [62], uncovering pieces of information in the data space that would otherwise pass unnoticed.

Visualizing the Document Ranking. Tags manipulated by the user are forwarded as parameters to the *Ranking Model*, which in turn feeds the ranking visualization. This visualization consists of a list of document titles (Figure 3(C)) and stacked bar charts (Figure 3(D)) depicting relevance scores for documents and keywords within them.

We favor the use of animation to convey ranking-state transitions rather than abrupt static changes. Animated transitions are inherently intuitive and engaging, giving a perception of causality and intentionality [20]. In *uRank*, soft animated transitions for ranking-state changes and document selection are meant to help the user intuitively switch contexts. As Baldonado et al. [61] state in their rule of attention management, perceptual techniques lead the user’s attention to the right view at the right time.

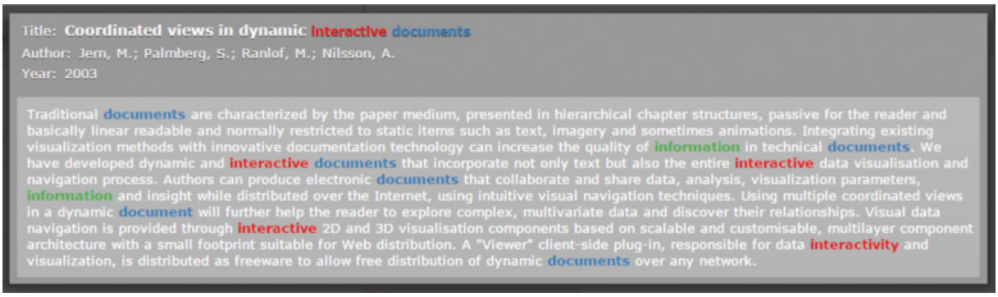
Once the document ranking is updated, the *Document List* is re-sorted in descending order by overall score and list items are translated to their new positions at a perceptible pace. In the *Ranking View*, stacked bars appear growing from left to right and horizontally aligned to each list item. Green or red shading effects are applied on the left side of list items for a few seconds to denote positive and negative shifts, respectively.

The total width of stacked bars indicates the overall score of a document, and bar fragments represent the individual contribution of keywords to the overall score. Bar colors match the color encoding for selected keywords in the *Query Box*, enabling the user to make an immediate association between keyword tags and bars. Missing colored bars in a stack denote the absence of certain words in the document surrogate. Additionally, each item in the *Document List* contains two types of numeric indicators: position and shift with respect to the immediate previous position. Position is denoted inside a gray circle, whereas shift is indicated as a color-coded number.

This visualization attempts to attract the user’s attention to likely relevant documents by bringing highly ranked ones to the top and pushing the rest to the bottom. Optionally, the user can track shifts in particular documents by clicking on the watch (eye-shaped) icon. The watched item remains in focus as it is surrounded with a slightly darker shadow and the title is underlined. Also, watched items remain on top of other elements during animated transitions.

Details on Demand. Once the user identifies documents that seem worth further inspecting, he or she can drill down one by one to determine whether the initial assumption holds. The *Document Viewer* (shown in Figure 7) gives access to textual content such as title and snippet, as well as other available metadata for a particular document.

Query terms and their families of words (e.g., “interactive” and “interactivity” are grouped under the tag “interaction”) appear highlighted in the text following the same color coding for tags in the *Query Box* and stacked bars in the *Ranking View*. These simple visual cues pop out from their surroundings, enabling the user to preattentively recognize keywords in the text and perceive their general context prior to conscious reading.



Title: Coordinated views in dynamic Interactive documents
Author: Jern, M.; Palmberg, S.; Ranlof, M.; Nilsson, A.
Year: 2003

Traditional documents are characterized by the paper medium, presented in hierarchical chapter structures, passive for the reader and basically linear readable and normally restricted to static items such as text, imagery and sometimes animations. Integrating existing visualization methods with innovative documentation technology can increase the quality of information in technical documents. We have developed dynamic and interactive documents that incorporate not only text but also the entire interactive data visualisation and navigation process. Authors can produce electronic documents that collaborate and share data, analysis, visualization parameters, information and insight while distributed over the Internet, using intuitive visual navigation techniques. Using multiple coordinated views in a dynamic document will further help the reader to explore complex, multivariate data and discover their relationships. Visual data navigation is provided through interactive 2D and 3D visualisation components based on scalable and customisable, multilayer component architecture with a small footprint suitable for Web distribution. A "Viewer" client-side plug-in, responsible for data interactivity and visualization, is distributed as freeware to allow free distribution of dynamic documents over any network.

Fig. 7. *Document Viewer* shows augmented title and abstract for a selected document. Color-coded terms match the tags in *Query Box.*

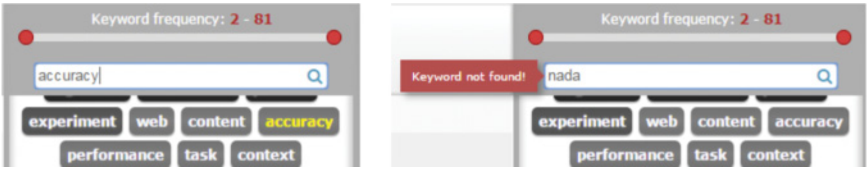


Fig. 8. Keyword search in *Tag Box.* (Left) Found keyword appears highlighted for a few seconds. (Right) Error message pops up otherwise.

3.2.4 Lessons Learned and Usability Improvements. In an earlier generation of *uRank*, the user had to click on the *document-* and *co-occurrence hint* to show effects (1) and (2) described in Section 3.2.1. The study reported in Section 4 revealed that clicking on the small icons representing the document and co-occurrence hints was offputting; therefore, we simplified this interaction by triggering their effects together on tag click.

We also found that users had difficulties finding a particular keyword, especially when the *Tag Box* was overly populated. For that reason, we added a text input field and a frequency range slider at the top of the *Tag Box*. The former provides keyword search functionality, such that if the term is found, the corresponding tag is highlighted and the *Tag Box* scrolls to its position; otherwise, an error message pops up (Figure 8). The frequency slider allows for setting the minimum and maximum document frequency for visible tags. Tags become visible or hidden as soon as the user starts dragging the handles. By default, the minimum value is set to 2 and the maximum value matches the most frequent tag. In Figure 3(A), the slider has been set to the range $[7, 81]$, significantly reducing the number of tags in display.

Additionally, dragged tags were previously removed from the *Tag Box*, causing position adjustments in subsequent tags to fill the empty gaps. We learned from user feedback that this had the undesired effect that already spotted tags could not be found again in the same place. Therefore, in the current version, a clone of a dropped tag remains in its original position and the border is highlighted with the same color assigned to the tag dropped in the *Query Box* (Figure 4(e)).

Items with null score were then hidden and the list height was shrunk or enlarged to fit only ranked items. Some users argued that such behavior was undesirable or confusing. Hence, unranked items are currently grouped below the ranked ones, so that they remain accessible even though they are likely irrelevant for the given query.

3.3 Analytical Methods

Control features in *uRank* (i.e., interactions with tags) are supported by a combination of text-analytic methods. We first extend the original documents with document vector representations containing their representative keywords, identify a set of global keywords thereof, and use them to calculate a query-document similarity ranking upon user interactions.

3.3.1 Keyword Extraction.

The described interactive features are supported by a combination of well-known text-mining techniques that extend the original documents with document vectors and provide meaningful terms to populate the *Tag Box*.

Document vectors ideally include only content-bearing terms like nouns and frequent adjectives, appearing in at least 25% of the collection; hence, it is not enough to just rely on a list of stop words to remove meaningless terms. First, we perform a part-of-speech tagging (POS tagging) [7] step to identify words that meet our criteria (i.e., common and proper nouns and adjectives). Filtering out nonfrequent adjectives requires an extra step. Then, plural nouns are converted into their singular form, proper nouns are kept capitalized, and terms in uppercase remain unchanged (they normally represent acronyms, e.g., "IT," "AR"). Porter stemming [41] is applied over the resulting terms, in order to increase the probability of matching for similar words (e.g., "robot," "robots," and "robotics" all match the stem "robot"). A document vector thus consists of stemmed versions of content-bearing terms.

Next, we generate a weighing scheme by computing *tf-idf* (term frequency–inverse document frequency) scores for each term in a document vector. This score is a statistical measure of how important the term is to a document in a collection. Therefore, the more frequent a term is in a document and the fewer times it appears in the corpora, the higher its score will be. Documents' metadata are extended with these weighted document vectors.

To fill the *Tag Box* with representative keywords for the collection set, all document keywords are collected in a global keyword set. Global keywords are sorted by document frequency (DF), that is, the number of documents in which they appear, regardless of the frequency within documents. To avoid overpopulating the *Tag Box*, only terms with DF above a certain threshold (default 5) are taken into account. The terms used to label keyword tags are actually representative words and not plain stems. Scanning a summary of stemmed words would turn unintuitive for users. Thus, we keep track of all variations matching each stem in order to allow for reverse stemming. The most representative term is then selected according to the following rules:

1. If there is only one term for a stem, use it to label the tag.
2. If a stem has two variants, one in lowercase and the other in uppercase or capitalized, use it in lowercase.
3. Use a noun ending in "ion," "ment," "ism," or "ty."
4. Use a term exactly matching the stem.
5. Use the shortest term.

To feed document hints, *uRank* employs a keyword-document matrix that links bearing documents to each global keyword. For co-occurrence hints (Figure 3.(A)), *uRank* builds a keyword-keyword matrix that tracks keyword co-occurrences with (by default) a maximum word distance of 2 (excluding stop words) and at least 10 repetitions across all documents.

3.3.2 Document Ranking Computation.

Quick content exploration in *uRank* depends on its ability to readily re-sort documents according to changing information needs. As titles and snippets are the only content available in document surrogates, we compute relevance scores with a term-frequency scheme. Term distribution schemes are rather adequate for long or full texts and are hence out of our scope. Boolean models have the disadvantages that they not only consider every term equally important but also produce absolute values that preclude a document ranking.

As the user manipulates keyword tags and builds queries from a subset of the global keyword collection, the *Ranking Model* builds a vector space model to compute document-query similarity using the document vectors previously generated during keyword extraction. The contribution that each query term adds to the document score should be clear in the visual representation, in order to give the user a transparent explanation as to why a document ranks in a higher position than another. However, a single relevance measure like cosine similarity alone is not enough to convey individual query-term contributions to the overall score. This is not a trivial issue, since the best overall matches are not necessarily the ones in which most query terms are found [18, 31]. Therefore, we break down the cosine similarity computation and obtain individual scores for each query term, which are then added up as an overall relevance score. Additionally, assuming that some keywords take precedence over others, the algorithm also takes into account weights specified by the user through tag sliders in the UI.

Given a collection of documents D and a set of weighted query terms Q selected by the user, the relevance score for document $d_i \in D$ is calculated by Equation (1):

$$s(d,Q) = \frac{1}{\mid Q \mid} \sum_{t \in Q} \frac{tfidf(t,d_i) \ \cdot \ \hat{w}_t}{\parallel d_i \parallel \cdot \parallel Q \parallel}, \tag{1}$$

where $tfidf(t,d_i)$ is the tf-idf score for term t in document d_i and $|| \ d_i \ | |$ is the Euclidean norm for vector d_i. Note that every term t in query Q is 1, such that $1/ \ ||Q| \ |$ represents a single dot in its unit query vector. Thus, the Euclidean norm of Q equals the square root of the vector length, that is, $|| \ Q \ | | = \sqrt{| \ Q \ |}$. In turn, $\hat{w}_t$ is the adjusted value for w_t, the weight assigned by the user to term $t \in Q$, such that $\forall t \in Q : 0 \leq w_t \leq 1$. $\hat{w}_t$ is obtained as shown in Equation (2):

$$\hat{w}_t = \frac{w_t}{\sum\limits_{t' \in Q} w_{t'}}, \quad s.t. \sum_{t \in Q} \hat{w}_t = 1. \tag{2}$$

The collection D is then sorted in descending order by overall score with the quicksort algorithm. Ranking positions are assigned, taking into account that documents with equal scores share the same rank. For example, if the first, second, and third elements in the sorted collection have equivalent scores, they are all assigned the first position, and the fourth item is assigned position number 2. Equation (3) describes the steps for rank assignment:

$$r_i = \begin{cases} 1 & \text{if } s(d_i,Q) = \max(s(d_k,Q)); \ \forall d_k \in D \\ r_{i-1} & \text{if } s(d_i,Q) = s(d_{i-1},Q) \\ r_{i-1} + \#D_p & \text{if } s(d_i,Q) < s(d_{i-1},Q), \end{cases} \tag{3}$$

where $D_p = \{d_j \in D \mid s(d_j,Q) > s(d_i,Q), \ r_j = r_{i-1}, \ 0 \leq j \leq i\}$ is a subset of documents in D immediately preceding d_i in the sorted list. Finally, shifts with respect to the position in the previous ranking state are calculated. The shift for items with previous null scores are indicated as "new" in the UI.

3.4 Implementation

uRank is a web-based tool entirely implemented in *JavaScript*. *jQuery*[1] and *d3*[2] are the core third-party libraries leveraged for the graphic design and interactions in the UI. Keyword extraction is performed on the client side (due to project requirements). POS-tagging, tokenization, stemming, and tf-idf computation are performed with the *jspos*[3] and a customized version of *NaturalJS*[4]

[1] https://jquery.com./

[2] http://d3js.org/.

[3] https://code.google.com/p/jspos/.

[4] https://github.com/amitamb/NaturalJS.

libraries. Other support components include *colorbrewer*,[5] for categorical and sequential color schemes, and *Underscore.js*,[6] for diverse functionalities.

A website[7] including an introductory video, a test version of uRank that works out of the box, and the link to the code repository is publicly available. Additionally, uRank can be installed as a *Bower*[8] component (`bower install --save urank`) and included as a third-party library in other web-based projects.

uRank is currently able to effectively display up to roughly 700 documents without considerable delays. In order to extend its capabilities and boost not only performance but also the user experience, we are in the process of developing a dedicated website, including user and collection management features. Nonetheless, an earlier version of uRank is currently integrated as a visualization component[9] in the EEXCESS Chrome extension.[10] As the user navigates any website, this extension pervasively scans its content and communicates with a federated system (F-RS) that in turn generates cultural and scientific recommendations from a variety of content providers (e.g., *Mendeley*,[11] *bit media*,[12] *ZBW*[13]).

4 EXPERIMENT I: URANK VERSUS LIST-BASED UI

Exploratory search interfaces have arguably a steep learning curve that often prevents their adoption. The goal motivating this study was to find out how people responded when working with a tool like *uRank* with respect to a traditional list-based UI, in terms of workload, completion time, and performance.

Additionally, we were interested in observing how our cognitive bottom-up approach elicits knowledge. Precisely, this evaluation motivated the construction of topic-oriented document collections related to a given query of piece of text. The items collected with both tools were then compared to the content-based ranking described in Section 3.3.2 and state-of-the-art topic analysis methods (LDA).

4.1 Methodology

The study was structured in a within-subjects design where participants performed four iterations of the same task with either *uRank* (U) or a baseline list-based UI (L) with usual web browser tools like *Control+F* search. Furthermore, we were interested in assessing the implications of exposing participants to pools of different sizes. We controlled this contextual aspect by introducing two variations of collection sizes, namely, 30 and 60 items. Therefore, the study followed a 2 × 2 repeated measures design with *tool* and *#items* as independent variables, each with two levels (*tool* = U/L, *#items* = 30/60). To counterbalance learning effects, we prepared four datasets covering a variety of topics and *topic* was treated as a random variable within constraints.

Every participant had to perform one task for each combination of the independent variables, that is, **U-30**, **U-60**, **L-30**, and **L-60**. All variable combinations were randomly assigned across participants with balanced Latin Square.

[5] http://colorbrewer2.org/.
[6] http://underscorejs.org/.
[7] http://urank.know-center.tugraz.at.
[8] https://bower.io/.
[9] http://eexcess.eu/visualisations/.
[10] http://eexcess.eu/results/chrome-extension/.
[11] http://dev.mendeley.com/.
[12] http://www.bit.at/.
[13] http://www.zbw.eu/.

A typical task unfolded in an exploration scenario in which the participant receives a list of recommendations while reading the *Wikipedia* page corresponding to the assigned topic. Note that participants were not actually exposed to any *Wikipedia* article. Instead, the task setup simulated a situation in which the user has already received a list of recommendations and starts to explore its content.

The general task goal was to "find the five most relevant items" for a given piece of text. In turn, each task comprised three subtasks (Q1, Q2, and Q3). On one hand, Q1 and Q2 targeted a specific focused search, and thus we supplied the participant with two or three search terms. Q3, on the other hand, was designed as a broad-search subtask. Hence, we provided an entire paragraph extracted from the *Wikipedia* article so that users had to decide themselves which keywords described the topic better. The motivation to ask for the "most relevant" documents was to avoid careless selection.

The system recorded the relevant items selected by participants across each task and subtask, as well as completion time for the overall task and for every individual subtask. Subjective feedback was collected through a posttask questionnaire consisting of a 7-point Likert *NASA TLX*[14] scale covering six dimensions of workload, namely: mental demand, physical demand, temporal demand, effort, frustration, and perceived performance.

We report the results for workload (Section 4.2.1) and completion time measures (Section 4.2.2) based on their respective multilevel models. For performance analysis, we investigated similarities and correlations in the lists produced for each subtask (Q1, Q2, Q3) per topic (Section 4.2.3). Finally, we built topic models with a state-of-the-art method (LDA) and contrasted them against the collections gathered with both tools (*U* and *L* conditions), as well as the ranking algorithm employed by *uRank* (Section 4.2.4).

4.1.1 Datasets. The datasets used in this evaluation covered a spectrum of cultural, technical, and scientific content, namely: *Women in workforce* (WW), *Robots* (Ro), *Augmented Reality* (AR), and *Circular economy* (CE). The main selection criterion was that *Wikipedia* provided a well-defined article for each one of them.

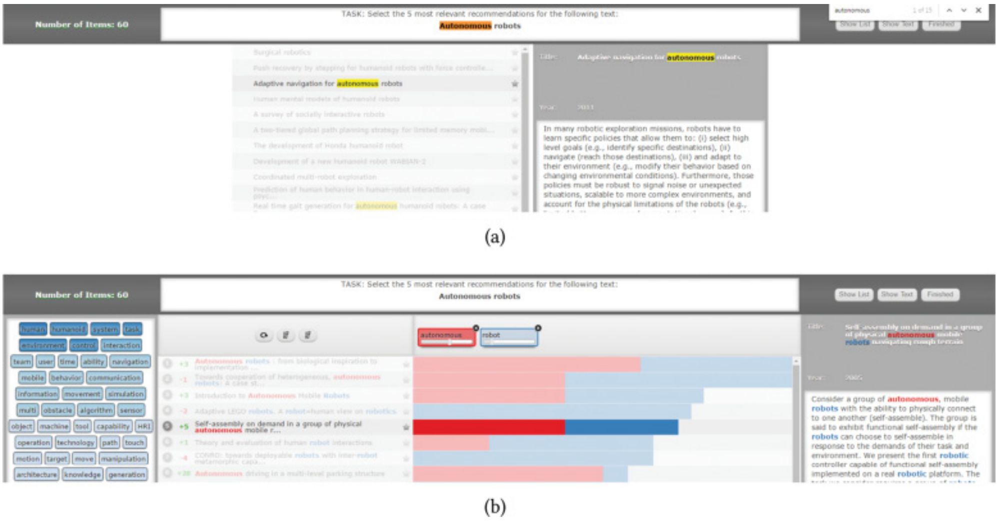
We prepared static lists of 60 and 30 items for each topic by feeding portions of texts from the original *Wikipedia* articles to the F-RS (see Section 3.4), which in turn preprocessed the text and queried a number of content providers. The result was a sorted merged list of items from each provider with no scoring information.

4.1.2 Participants. A total of 24 participants took part in the study (*11* female, *13* male, between 22 and 37 years old). They were recruited from the medical and computer science university population. None of them was knowledgeable in the topic areas selected for the study.

4.1.3 Apparatus/Materials. The study was run on an Apple MacBook Pro (2.7MHz i7 Quad-Core) with 15-inch retina display (2880x1800). The service was run in a local server, and the study interfaces were presented in a Google Chrome browser.

4.1.4 Procedure. At the beginning of the study session, the participant had to answer a few questions about demographic aspects. Then they were presented with an introductory video explaining the functionality of *uRank*, aiming for all participants to receive exactly the same instructions and explanations. Next, they performed a short training with a different topic (*Renaissance*) to familiarize with both *uRank* and the baseline tools. At the beginning of the first task, the system showed a short text describing the topic and the task to be fulfilled. After reading the text, the participant pressed "Start" to redirect the browser to the corresponding UI. Figure 9(a) illustrates

[14] http://humansystems.arc.nasa.gov/groups/tlx/.



(a)

Number of Items: 60

TASK: Select the 5 most relevant recommendations for the following text:
Autonomous robots

Surgical robotics
Push recovery by stepping for humanoid robots with force controlle...
Adaptive navigation for autonomous robots
Human mental models of humanoid robots
A survey of socially interactive robots
A two-tiered global path planning strategy for limited memory mobi...
The development of Honda humanoid robot
Development of a new humanoid robot WABIAN-2
Coordinated multi-robot exploration
Prediction of human behavior in human-robot interaction using psyc...
Real time gait generation for autonomous humanoid robots: A case

Title: Adaptive navigation for autonomous robots
Year: 2011

In many robotic exploration missions, robots have to learn specific policies that allow them to: (i) select high level goals (e.g., identify specific destinations), (ii) navigate (reach those destinations), (iii) and adapt to their environment (e.g., modify their behavior based on changing environmental conditions). Furthermore, those policies must be robust to signal noise or unexpected situations, scalable to more complex environments, and account for the physical limitations of the robots (e.g.,

(b)

Number of Items: 60

TASK: Select the 5 most relevant recommendations for the following text:
Autonomous robots

+3 Autonomous robots : from biological inspiration to implementation ...
-1 Towards cooperation of heterogeneous, autonomous robots: A case st...
+3 Introduction to Autonomous Mobile Robots
-2 Adaptive LEGO robots. A robot+human view on robotics
+5 Self-assembly on demand in a group of physical autonomous mobile r...
+1 Theory and evaluation of human robot interactions
-4 CONRO: towards deployable robots with inter-robot metamorphic capa...
+28 Autonomous driving in a multi-level parking structure

Title: Self-assembly on demand in a group of physical autonomous mobile robots navigating rough terrain
Year: 2005

Consider a group of autonomous, mobile robots with the ability to physically connect to one another (self-assemble). The group is said to exhibit functional self-assembly if the robots can choose to self-assemble in response to the demands of their task and environment. We present the first robotic controller capable of functional self-assembly implemented on a real robotic platform. The

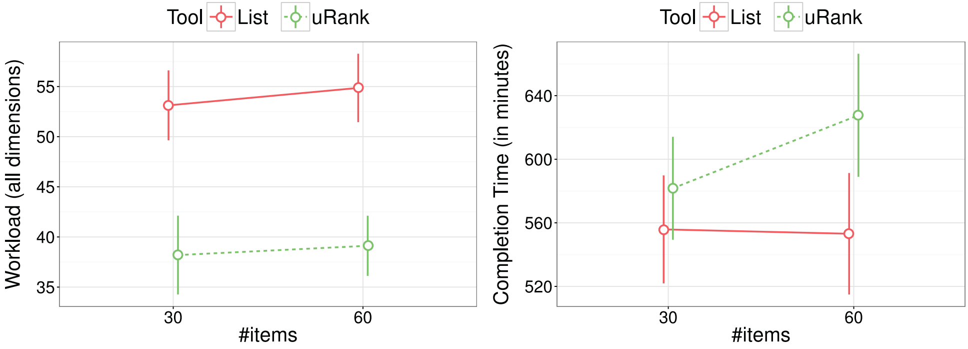
Fig. 9. Cropped screenshots of the user interfaces employed in study I. (a) Baseline list-based UI. (b) uRank's early version.

the list-based UI for the baseline condition, while Figure 9(b) portrays the early version of uRank's UI used at the time. At this point, the first subtask of task 1 began and the internal timer initiated the count without disturbing the user. The goal of the subtask and the reference text was shown at the top of the UI (see Figure 9). Participants were able to select their relevant items by clicking on the star-shaped icon and inspect them later on a drop-down list (visible by clicking on "Show List"). At the beginning of each subtask, the items in the collection were ordered randomly, ensuring that an item would not appear in the same position in the next subtask.

During the study, the experimenter reminded the participants when the allotted time was almost over but did not force them to abandon the task (see Section 4.1.5). The subtask was concluded by clicking on the "Finished" button and then the second subtask started immediately. The UI alerted participants when attempting to finish without collecting five items but allowed them to continue. The second and third subtasks proceeded in like manner. After the three subtasks concluded (i.e., the whole task), participants had to fill the *NASA TLX* questionnaire to assess workload under the given *tool* and *#items*. The procedure for the remaining tasks was repeated following the same steps. Finally, participants were asked about comments and preferences.

4.1.5 Pilot Study: Time Constraints. Three participants took part in a pilot study. None of them was familiar with the tool or topics in the experiment. After this preliminary stage, we realized that having asked for "the most relevant items" made the experiment overly long, as participants tried to carefully inspect their selections (particularly in the *List* condition). Based on the pilot, we decided to constrain the duration of the three subtasks Q1, Q2, and Q3 to 3, 3, and 6 minutes, respectively.

The time constraint was albeit not a hard deadline. In the actual study, the experimenter told participants they had reached the allotted time but did not prevent them from continuing with the task. The constraint accomplished two goals: (1) it introduced stress by having to complete the task in a short (but manageable) time, and (2) it allowed us to study strategies under such conditions. During the pilot, having an open-ended study made participants take time to carefully read each item before making a decision, clearly just because of the nature of the formal study. Setting a time limit in the end made for a closer-to-real condition.



(a) (b)

Fig. 10. Study I results. (a) Workload interaction lines show that *uRank* is significantly less demanding. (b) Time completion lines show a tendency toward using all available time.

4.2 Results

The study produced a total of 96 observations grouped according to 24 participants that worked under the four possible treatments. Hence, the analysis of workload and temporal aspects in the context of our repeated measures design is based on fitted multilevel models (MLMs) with random intercepts, where *participants* are treated as a random effect and the independent variables *tool* and *#items* as fixed effects. Sections 4.2.1 and 4.2.2 report on these results. Moreover, we investigated similarities and correlations in the lists produced for each subtask (Q1, Q2, Q3) and topic (Section 4.2.3). Finally, we disclose an empirical analysis of the collections gathered with both tools against topic models created with state-of-the-art methods (Section 4.2.4).

4.2.1 Workload. We estimated workload by fitting a linear multiple mixed-effects model using maximum likelihood, such that the predictors *tool* and *#items* are fixed effects nested within the random effect *participant*.

In order to observe whether individual effects improve the model, we first fit a baseline MLM (or "empty" model) that includes no predictors other than the intercept, and then build up the model therefrom adding one effect at a time. Intuitively, the variable of most interest is *tool*, and thus it is the first main effect added to the model. Then follows the main effect of *#items* and lastly the interaction between the two. Likelihood ratio tests are reported on each step, as well as the t-test against the baseline condition ($U vs. L$) and the corresponding fixed coefficient.

The baseline model is significantly outperformed when *tool* is added as a predictor, $\chi^2(1) = 40.60,\ p < .0001$, which means that *tool* is a good predictor for *workload*. The model predicts a lower overall workload working with *uRank* than with the *List*-based benchmark tool, $b = -7.67,\ t(23) = -7.25,\ p < .001,\ 95\%\ CI = [-10.30, -5.05]$. The slope b estimates a decrease of 7.67 points in the NASA TLX scale (recall it is adjusted to a maximum score of 100).

Conversely, accounting for the effect of *#items* does not yield a better model fit, $\chi^2(1) = 0.41,\ ns$. Indeed, duplicating the item pool size from 30 to 60 increased workload only marginally, $b = 0.67,\ t(46) = 0.63,\ ns$. This may indicate that a pool of 60 items is not large enough for the user to perceive additional difficulties. Perhaps a more appropriate growth to detect such effect would be, for example, from 30 to 100 documents.

The interaction of the main effects did not improve the goodness of fit either, $\chi^2(2) = 0.04,\ ns$. This is not surprising, as it can be observed in Figure 10(a) that although the gap between *List* and *uRank* seems rather large at both 30 and 60 items, the interaction lines show a slight growth but remain parallel. It is worth pointing out though that workload measurements remained lower for *uRank* with 60 items compared to the *List* with 30 items. Post hoc multiple comparisons for the general linear hypothesis with Tukey contrasts revealed that the difference for $L - 30 vs. U - 60$ was significant at $p < .001,\ CI = [-21.52, -6.48]$.

We further broke down the interaction and fit a model with observations under 30 and 60 items separately (with tool as a unique predictor). Better goodness of fit was achieved with the 60-item subset, in contrast to the 30-item data. Since the identical model was fitted with two different datasets, no likelihood ratio can be estimated. However, knowing that the model can be more accurate under these conditions should be taken into consideration for further studies.

The analysis can be extended to all workload dimensions. In all cases, *tool* turned out to be a good predictor (all chi-squared values significant at $p < .0001$), while *#items* did not report any fit improvements. Refer to Appendix A.1 for further details about the distribution of responses for each individual workload dimension. Appendix B provides full disclosure about the MLM employed in the current analysis.

4.2.2 Completion Time. We built similar MLMs with the same structure as described earlier but computed for the overall time outcome and the individual times recorded per task. The model for overall completion time yielded a better fit when the *tool* predictor was added to the baseline model, $\chi^2(1) = 4.56,\ p < .05$. *#items* and the interaction of the main factors did not provide any significant improvement to the model.

We found that participants took on average approximately 50 seconds longer to complete a full task (adding times for Q1, Q2, and Q3) with *uRank*, $b = 25.10,\ t(23) = 2.16,\ p < .05$. This difference is solely explained by the time recorded for Q3, $b = 23.65,\ t(23) = 2.49,\ p < .05$, that is, searching for items related to a full paragraph rather than a couple of keywords. The *#items* variable was in turn a good predictor for time on the second focused task (Q2): participants took longer finding relevant items in the longer list, $b = 8.25,\ t(46) = 2.04,\ p < .05$.

Although we intentionally introduced a certain level of stress by setting (flexible) time constraints, the fact that tasks developed with *uRank* took longer is not necessarily a negative or undesired outcome, especially if perceived time pressure is brought into consideration. Temporal demand measures indicate that participants felt significantly less pressed to finish while performing with our tool, $b = -0.43,\ t(23) = -3.81,\ p < .001,\ 95\%\ CI = [-0.65, -0.20]$. In the closing interviews, most participants admitted to leveraging the extra time to carefully make sure they had the right items. They claimed that some tasks were particularly hard to solve when the search terms rarely appeared in titles. Hence, with *uRank,* they repeatedly had to re-sort the results, then quickly scan titles and inspect the abstracts of the most promising ones. In the case of the *List*, they tended to be less thorough and select items at first glance.

4.2.3 Performance. Assessing performance is not a trivial task, as relevance is a rather subjective measure. In the absence of some ground truth and considering a possible bias that could have been introduced if we had recruited a small group of experts on each topic to define it, we decided instead to analyze "consensus" in the collections generated by the participants. Thus, we aggregated the collections gathered under all the manipulated conditions and computed cosine similarity across *tool*, *#items*, topic (WW, Ro, AR, and CE), and subtask (Q1, Q2, and Q3).

Similarity values between collections produced with *uRank* and the list-based UI across all subtasks (U vs. L) denote that participants' choices regarding relevant documents matched approximately three out of four times ($\mu = .73,\ \sigma = .1$), regardless of the number of items to which they were exposed.

Table 1 shows that collections produced with our tool for the two variations of *#items* (U-30 vs. U-60) turned highly similar regardless of topic and subtask ($\mu = .80,\ \sigma = .12$, with a minimum

Table 1. Cosine Similarities Between Collections Gathered During Study I

Task Type	Comparison	WW	Ro	AR	CE	All Topics
Q1 (focused search)	U vs. L	.55	.79	.58	.74	.66
	U-30 vs. U-60	**.71**	**.83**	**.94**	**.67**	**.79**
	L-30 vs. L-60	.58	**.83**	.56	.56	.63
	ICC	.66	.86	.81	.91	–
Q2 (focused search)	U vs L	.70	.86	.84	.86	.81
	U-30 vs. U-60	**.84**	**.89**	**.90**	**.93**	**.89**
	L-30 vs. L-60	.82	.74	.81	.87	.81
	ICC	86	.84	.87	.87	–
Q3 (broad search)	U vs. L	.75	.72	.75	.63	.72
	U-30 vs. U-60	**.64**	**.88**	**.75**	**.62**	**.72**
	L-30 vs. U-60	.59	.66	.63	.33	.55
	ICC	.75	.74	.81	.71	–

of .62). Comparisons for the *List* with 30 and 60 items (L-30 vs. L-60) denote greater diversity ($\mu = .67,\ \sigma = .16$, with a minimum of .33). Not surprisingly, similarity values tend to decrease for broad search task (Q3) ($\mu = .66,\ \sigma = .13$) compared to focused search (Q1 and Q2) ($\mu = .77,\ \sigma = .13$).

Summarizing, the results suggest that with our tool, users tend to make uniform decisions, as the number of items to which they are exposed grows. Nevertheless, the proportion of matching documents in list-generated collections (two out of three) still conveys a moderate consensus. The decrease in consensus for broad searches with respect to focused tasks could be explained by the variability across participants at the moment of choosing the search terms for a given text larger than a couple of words.

Additionally, we aggregated the items collected with the list (L_p) and *uRank* (U_p) for each topic and compared them with the scores received by our ranking algorithm (*CB*). We performed intra-class correlations (ICC) with a two-way, consistency, average measures model and found good to excellent ICCs across all subtasks, namely: on average, .81 for Q1, .86 for Q2, and .75 for Q3. Results are summarized in the last row of each subtask in Table 1. A closer look at the distribution of scores in Figure 11 reveals that highly ranked documents by *CB* were also popular choices within U_p and L_p. For focused searches (Q1 & Q2) *CB*, U_p, and L_p share at least five popular items across all topics (aligned blocks with high intensity along rows 2, 3, and 4 in Q1 and Q2 heatmaps for each topic). In the case of broad searches (Q3), L_p produced more widespread collections with less individual favorites, while U_p yielded more consensus in its collections (more blocks with higher intensity). The overlap between *CB*, U_p, and L_p was hence moderate, with roughly three items.

Overall, we found a high consensus in collections gathered with both tools, albeit higher in those collected with *uRank*. With the *List*, having more items to search through resulted in lower consensus. The preference for *uRank* was mostly reflected in a more satisfactory user experience. Results for temporal demand reported in Section 4.2.1 and measures for perceived performance ($b = -0.45,\ t(23) = -3.44,\ p < .01,\ 95\%\,CI = [-0.71, -0.18]$) support this postulate. Nevertheless, despite the overall consensus achieved with both tools, collections gathered with *uRank* were less susceptible to growths in the dataset size (U-30 vs. U-60 and L-30 vs. L-60 comparisons in Table 1). Intuitively, exposing participants to larger result lists would likely yield fewer overlapping most popular items for the baseline tool.

4.2.4 Emergent Topics. Having defined collections and subtopics, we created topic models from the collections as baseline state-of-the-art topic analysis. To generate the topic models, the

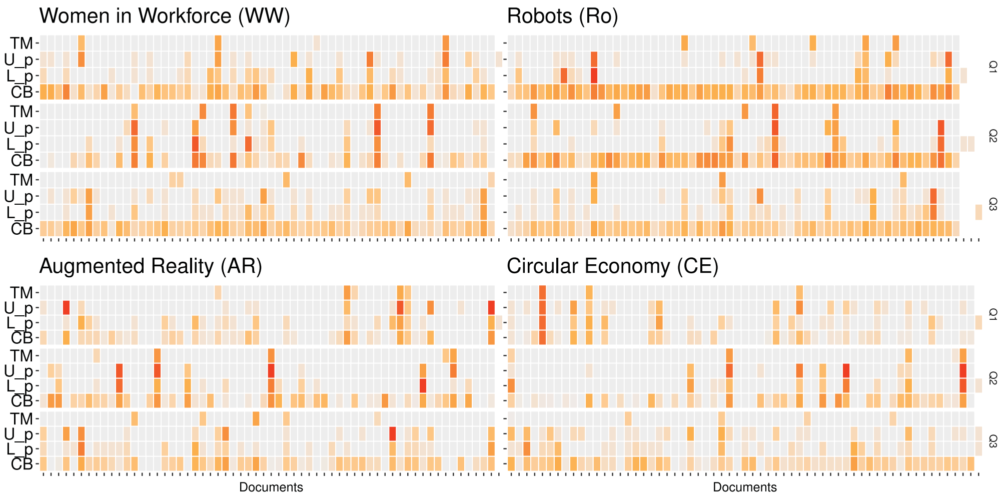


Fig. 11. Correlations heatmap per topic. Most popular items collected with our tool (U_p) had high scores in the *CB* ranking. Most popular items collected with the list (L_p) are more widespread. Documents found in the topic model (*TM*) also had high *CB* rank and were picked often (U_p, L_p).

topicmodels package of R was used, with variational expectation-maximization and estimated α. Finding the right parameters for topic modeling was a challenge. In the end, a topic model was created for each collection, with parameters ($K = 30$, $\alpha = estimated$) so that the focused exploration questions (Q1 and Q2) were covered by one or more topics. The broad exploration questions (Q3) were adjusted by choosing the highest number of words in the subject text appearing in a single topic. Document scores within each topic were modulated with cumulative *tf-idf* scores for the chosen keywords.

Admittedly, attempting to recreate information needs along the division of topic models was a daunting task; it required several iterations of adjusting parameters, recreating the model, and searching through generated topics. Configurations that could answer one question were not able to answer the next one, due to fragmented topics. Our approach interactively accomplished similar results, with the added value that the cognitive experience remains in the user.

The topics obtained with topic modeling (TM) were used as a basis for comparisons against the *CB* ranking and the collections of documents generated by the 24 participants with the *List* (L_p) and *uRank* (U_p). To obtain normalized scores for the most popular items in each collection, we used frequency of choice (times chosen/trials, where trials = 12). Figure 11 illustrates the overlap in score results. Topic models (*TM*) were very narrow, while the *CB* ranking tended to produce widespread results with low peaks when the requirements used many keywords. It seemed that there were common high-scoring documents highlighted in the topic and also popular among participants. To establish whether there is a valid correlation, we first collected the top five scoring items for each method (CB, U_p, L_p, TM) and formed a matrix. The dimensions of the matrix varied due to the nonperfect overlap among top-scoring items across methods. We collected the scores from each method for all the documents in the matrix and computed *Pearson* product-moment correlations with TM: ($CB - TM$, $U_p - TM$, $L_p - TM$). Table 2 summarizes the correlation results.

It is important to highlight that the scoring mechanisms in topic models are different from the one used by the *CB*ranking and the one for the most popular items collected with both tools (U_p and L_p). Despite differences in scoring schemes, Table 2 shows a number of significant correlations.

Table 2. Emergent Topics

	Q1	Q2	Q3
		WW	
CB	$r_{(11)} = -.00, p = .99$	$r_{(6)} = .42, p = .29$	$r_{(11)} = -.15, p = .60$
U_P	**$r_{(11)} = .68, p < .05$**	$r_{(6)} = .39, p = .32$	$r_{(11)} = -.65, p = .05$
L_P	$r_{(11)} = .16, p = .59$	$r_{(6)} = -.55, p = .24$	**$r_{(11)} = .56, p < .05$**
		Ro	
CB	$r_{(11)} = -.41, p = .16$	$r_{(9)} = .00, p = .98$	**$r_{(11)} = .55, p = .05$**
U_P	$r_{(11)} = -.41, p = .16$	$r_{(9)} = .05, p = .86$	$r_{(11)} = .09, p = .74$
L_P	$r_{(11)} = -.52, p = .06$	$r_{(9)} = -.06, p = .80$	$r_{(11)} = -.35, p = .23$
		AR	
CB	$r_{(9)} = .38, p = .24$	$r_{(9)} = .06, p = .84$	$r_{(11)} = -.01, p = .95$
U_P	$r_{(9)} = .08, p = .80$	$r_{(9)} = .54, p = .08$	**$r_{(11)} = -.61, p < .05$**
L_P	$r_{(9)} = .01, p = .96$	$r_{(9)} = .00, p = .99$	**$r_{(11)} = -.58, p < .05$**
		CE	
CB	**$r_{(6)} = .82, p = .01$**	$r_{(8)} = .23, p = .51$	**$r_{(11)} = -.57, p < .05$**
U_P	$r_{(6)} = .64, p = .08$	$r_{(8)} = .30, p = .39$	$r_{(11)} = -.11, p = .7$
L_P	$r_{(6)} = .53, p = .17$	$r_{(8)} = .42, p = .22$	$r_{(11)} = -.24, p = .4$

Correlations: Top-5 ranked elements (CB), popular with *uRank* (U_P) and the *List* UI (L_P) compared to documents from the corresponding topic model. *Note:* Bold indicates significance at $p < .05$.

Finding correlations with the topic models was a surprise. Admittedly, topic models are used for exploration in a different way. Yet, this provides evidence that cognitive topic analysis led to comparable results. Due to its workflow of exploration and discovery, our approach lets the user take control over the organization of the corpus along his or her information needs.

4.3 Discussion and Limitations

It is a long-known problem that most users do not go beyond simple keyword search, in spite of the effort invested in more advanced UIs. Nonetheless, this study shed light on the benefits of a tool like uRank. Even though our tool is inherently more complex than a Google-like UI, users felt significantly less pressed by time and more confident about their performance. We also observed that documents highly ranked by the *CB* algorithm were frequently collected not only with *uRank* (which is the obvious case) but also with the *List*. Moreover, we observed many overlaps between items chosen by participants and those in the topic models. Lower levels of workload and similar bookmark choices suggest that *uRank* effectively helped to alleviate the cognitive load without hindering performance. An important remark is that participants were first-timers and only had a few minutes of training prior to the evaluation tasks; hence, we would expect increased acceptance of the tool with further usage.

Admittedly, the highly controlled and rather unnatural setting posed certain limitations. For example, we balanced the level of knowledgeability by choosing topics in which none of the participants was an expert. Having little or no knowledge about the underlying topic made tasks quite difficult for some of them. Also, exploration was guided toward common "search interests," which probably did not represent the true interests of the users. Taking these limitations into account, we conducted a second user study to observe user behavior during exploratory search in more realistic conditions.

5 USER STUDY II: ACTION ANALYSIS AND USABILITY

This study was intended to analyze the exploratory search strategies users employ when working with *uRank* in a more natural setting, that is, without the shortcomings from study I, such as rigid tasks, time-ups, or uninteresting topics. We were particularly interested in addressing usability aspects and detecting usage patterns.

5.1 Methodology

This empirical study was conducted with the purpose of assessing in detail what components of *uRank* were more successfully used in a real search setting, specifically, within the framework of the conceptual model introduced in Section 3.1. Thus, we asked participants to freely explore a collection of documents for the topic of their selection and bookmark a few documents of interest. Meanwhile, the system collected logs for three kinds of actions: prediction, control, and drill-down.

Prediction actions are linked to the stage where the system tries to help the user become aware of possible exploration paths, that is, keywords of interest that would be potentially worth selecting. Control actions refer in general to ranking updates, where the user is exploring the document collection. Lastly, drill-down actions are related to the user seeking explanations that validate his or her initial assumptions about the relevance of a particular document. These results are reported in Section 5.2. Additionally, we split control action logs based on the moment of the first bookmark and analyzed whether this was actually a tipping point during search sessions (Section 5.3). We contrasted both analyses with subjective data collected at the end of the task. Finally, we report on subjective usability aspects measured with a standard scale.

5.1.1 Evaluation Protocol. We sent invitations via email to colleagues in the computer science field, which included a link to a web form with all necessary guidelines. A total of 24 people accepted to take part in the study. However, we only took 20 of them into consideration, as we presumed the remaining four dropped the study (they did not create any bookmark although it was clearly stated in the study guidelines).

A session started with a demonstrative video of the UI features. Thereafter, participants had to open the *uRank* website in their browsers. They could choose a topic of their preference out of seven collections, each with approximately 100 documents. The task consisted of freely exploring the collection and bookmarking interesting documents. We suggested between five and 10, although did not impose this as a strict condition.

The system recorded action logs, such as tag clicks, single and multiple drag-and-drop interactions, and so forth. After submitting the session data, users filled out a survey consisting of (1) questions addressing usability aspects of specific UI components and (2) a standard usability questionnaire.

5.2 Action Analysis

In this section, we break down action log information according to action-type categories: exploration, control, and drill-down, which in turn are intrinsically related to the three stages described in our model. Table 3 presents a summary of these recorded behaviors.

We contrast user behavior against subjective feedback provided for UI-specific questions. Responses were collected on a 7-point Likert scale (−3 = strongly disagree, 3 = strongly agree). Most questions were phrased in a positive tone, for example, “Looking at the colors in keywords and bars, it was clear how the ranking was computed.” The scale was inverted for negative-tone questions, for example, “The position indicator in the document list was confusing,” and hence, all values can be interpreted as “higher is better.” Figure 12 illustrates meaningful relationships extracted from both behavioral and subjective data, putting ranking updates (an aggregation of all

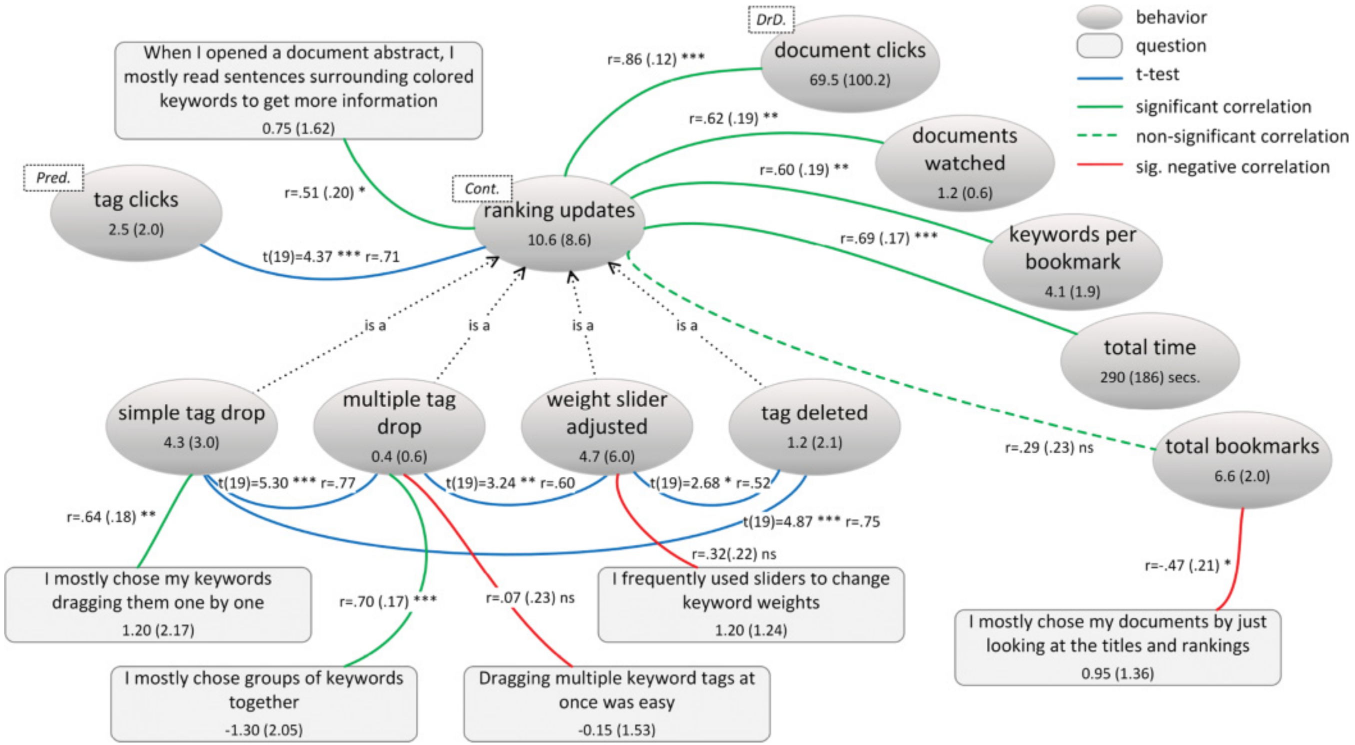


Fig. 12. Relationship network: t-tests and correlations between logged activity and subjective feedback collected during study II. The diagram shows that simple tag drops and weight changes were the most frequent types of raking updates and that the more updates there were, the more documents were inspected in detail (clicks and watches). Also, participants performing several updates tended to use more keywords per bookmark (number of tags in the *Query Box* at the moment of the bookmark) and spend more time exploring. Legends indicate the meaning of the different arrow types. Significance values are indicated as *** ($p < .001$), ** ($p < .01$), * ($p < .05$), and *ns* ($p \geq .05$). Correlations also report standard errors between parentheses.

Table 3. Cosine Similarities Between Collections Gathered During Study II

Type	**Action**	μ (σ)	**#Users**
Prediction	Tag hover	124.7 (19.6)	20
(*awareness*)	Tag click	2.5 (0.5)	20
	Keyword search	0.2 (0.2)	1
	Frequency range slider	24.2 (15.9)	4
Control	Ranking update	10.6 (1.9)	20
(*exploration*)	Single tag dropped	4.3 (0.7)	18
	Multiple tags dropped	0.4 (0.1)	7
	Tag weight changed	4.7 (1.3)	15
	Tag deleted	1.2 (0.5)	8
	Reset	0.2 (0.2)	1
Drill-down	Document click	69.5 (22.4)	19
(*explanation*)	Document bookmark	6.55 (0.5)	20
	Document unbookmark	0.1 (0.1)	1
	Document watched	1.2 (0.3)	13
	Document unwatched	0.3 (0.1)	5

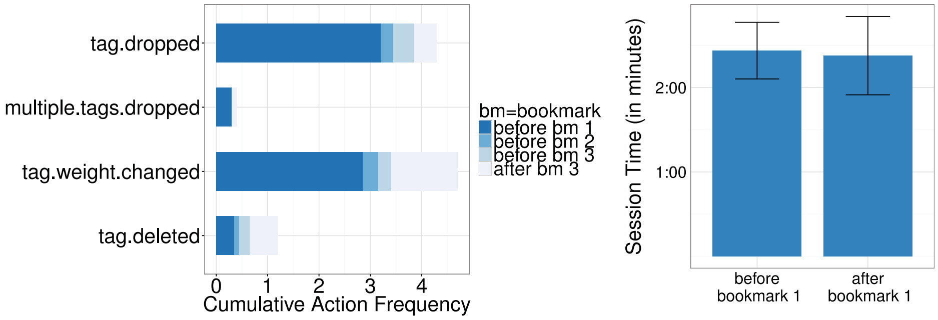
control actions) at the center of the network. In the following analysis, we often refer to these relationships to draw our conclusions.

5.2.1 Prediction Actions: Becoming Aware. Action logs show that users extensively hovered on keyword tags (124 times on overage) and additionally clicked on a tag around 2.5 times to preview bearing documents and co-occurring keywords. The fact that all participants performed tag clicks in *Tag Box* is a promising indicator for usefulness.

Usage data for the keyword range slider revealed that, despite positive answers stating it was useful to reduce the number of visible tags ($\mu = 1.05,\ \sigma = 1.76$), only four participants actually used it. The keyword search feature was also scarcely used: only one participant actively searched for specific terms. This corresponds with previous observations: people preferred browsing the offered keywords to explicitly searching for them [19].

5.2.2 Control Actions: Exploring by Ranking On the Fly. Ranking updates were significantly more frequent than tag clicks in the *Tag Box*, $t(19) = 4.37, p < .001$, which may suggest that users were more interested in immediately discovering the effect of manipulating a tag rather than trying to predict it beforehand. This evidences the role of interactive control features in engaging users in a trial-and-error search process.

Most control actions corresponded to simple tag additions (4.3 times) and weight changes (4.7 times). Measurements for simple tag selections correlate with user responses, as they admitted mostly choosing keywords one by one ($\mu = 1.20,\ \sigma = 2.16$), $r = .65, p < .05$. Multiple drag-and-drops were far more infrequent, $t(19) = 5.30, p < .001$, and only performed by a third of the participants (seven), which matches the disagreement with the question "I mostly chose groups of keywords together" ($\mu = -1.13,\ \sigma = 2.05$), $r = .70,\ p < .001$. When asked about how easy it was to use this mechanism, responses were quite neutral ($\mu = -0.15,\ \sigma = 1.53$). Although this answer does not correlate with usage logs, $r = .07,\ ns$, perhaps the interaction was indeed too complicated (two participants actively reported so). We attribute the low usage to its two-step mechanism, evidencing that a one-step increment in the interaction path inherently hinders ease of use.



(a)

(b)

Fig. 13. Study II Results. (a) Stacked bars for ranking updates distribution around first, second, and third bookmark. (b) Bar chart shows average pre- and post-first-bookmark session time.

As for changes to tag weights, 75% of the participants performed this action approximately 6.3 times (for $N = 15$) and indeed reported frequently using the sliders to change keyword weights ($\mu = 1.20,\ \sigma = 1.24$) (no correlation was found though). Tag deletions remained in a low rate, as two-thirds of the participants never performed this action. The rest did it between 3 and 4 times.

5.2.3 Drill-Down Actions: Seeking Explanations. All but one participant clicked several times on a specific document to access full abstracts an average of 73 times (with a 95% $CI = [24,\ 121]$ for $N = 19$). In general, participants admitted that they often read document titles ($\mu = 1.70,\ \sigma = 1.34$) and reported reading full abstracts only moderately ($\mu = 0.55,\ \sigma = 1.54$).

In turn, every participant bookmarked roughly six documents (95% $CI = [5.65,\ 7.55]$). We argue this is not discouraging, considering that the task did not impose a minimum amount. They did not revert their decisions, as we only accounted for one item that was accidentally unbookmarked and immediately bookmarked again. Participants acknowledged mostly choosing their bookmarks by looking at the titles and the ranking ($\mu = 0.95,\ \sigma = 1.36$). However, we found a negative correlation between the level of agreement with this statement and the total number of bookmarks, $r = -.47,\ p < .05$, which may suggest that more engaged users (more bookmarks) were indeed more careful choosing items. Their responses also revealed that they clearly understood how the ranking was computed by looking at the color of tags and bars ($\mu = 1.80,\ \sigma = 1.28$), denoting that the UI was able to convey a transparent logic. Interestingly, the agreement to often reading sentences surrounding colored keywords ($\mu = 0.75,\ \sigma = 2.62$) is higher for participants that bookmarked more items, $r = .51,\ p < .05$.

Summarizing, the evidence suggests that *uRank* helped participants achieve a high level of self-perceived performance and that they felt confident about their decisions.

5.3 Search Strategy Analysis

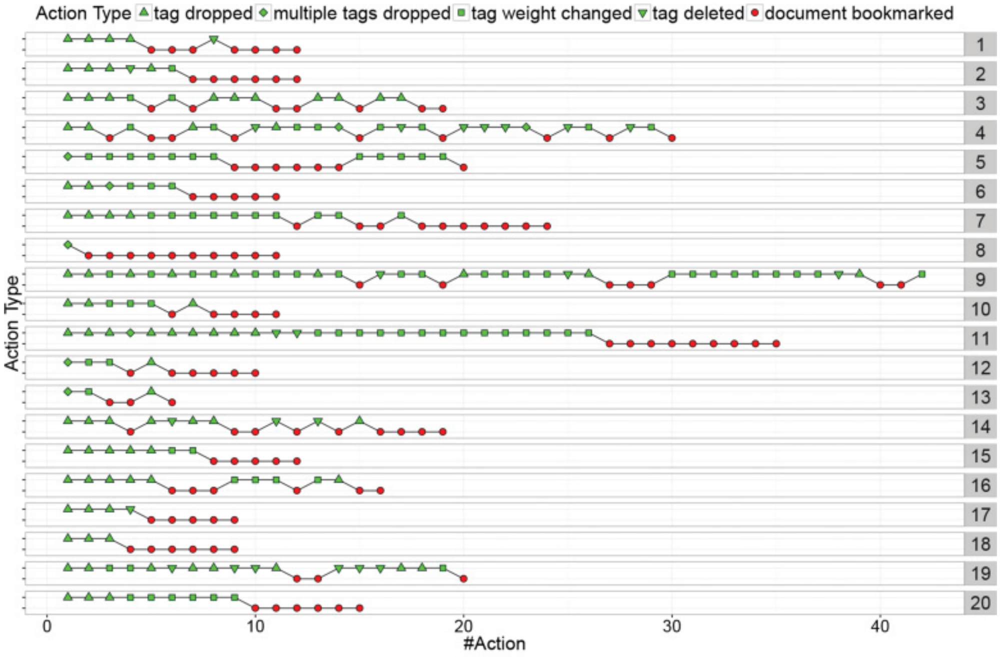
Control actions represent trial-and-error steps that users perform throughout browser-based exploration. We attempt to discover search patterns thereof by splitting ranking update actions, namely, *tag dropped*, *multiple tag dropped*, *tag weight change*, and *tag deleted*, and observing how frequently they occurred (1) before the first bookmark and (2) between subsequent bookmarks.

At first glance, Figure 13(a) reveals a strong tendency for tag selections to occur prior to the first bookmark event. A paired t-test confirms this observation, $t(19) = -3.46, p < .01$. Almost all multiple tag drops were performed at this stage, while roughly 75% of all single tag drops fell therein ($\mu = 3.2,\ \sigma = 2.33$). As stated before, tag deletions were infrequent, but appear slightly more frequently in the second half ($\mu = 0.35,\ \sigma = 0.81$ before and $\mu = 0.85,\ \sigma = 1.81$ after the first bookmark break). Weight changes also reach their peak before the first bookmark cut ($\mu = 2.85,\ \sigma = 3.73$), then tend to decrease toward the second ($\mu = 0.30,\ \sigma = 0.66$) and third bookmarks ($\mu = 0.25,\ \sigma = 0.91$), and finally a slight increment appears toward the end ($\mu = 1.30,\ \sigma = 2.70$). The distribution of actions depicted in Figure 13(a) suggests that after the first bookmark, users were quite certain about the chosen keywords, although they minimally fine-tuned keyword weights even after identifying three relevant documents.

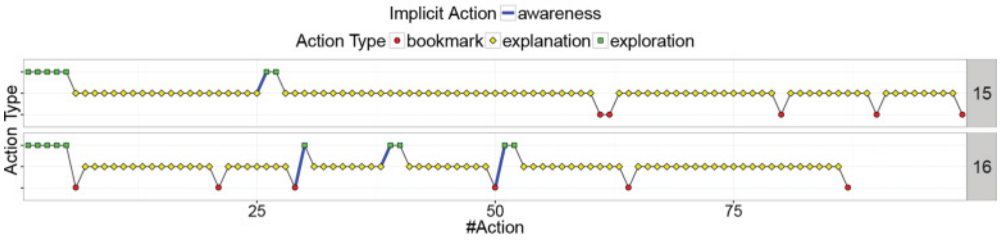
This notion is supported by the time distribution. With an average session of roughly 5 minutes, Figure 13(b) reveals that pre- and post-first-bookmark exploration were balanced (on average 2'26" and 2'22", respectively). This suggests that the tipping point in a session was almost exactly marked by the occurrence of the first bookmark. Control actions (i.e., ranking updates) in general were executed in the first half of a session. Thereafter, users minimally refined query parameters, denoting that they were satisfied with their decisions, and dedicated the second half of the session to find other relevant documents. Moreover, the fact that tag drops and weight sliders were extensively used and that tag deletions were scarce may indicate that participants tended to know what they were searching for from the beginning and rarely undid their decisions. Further validation can be found in the number of keywords per bookmark (amount of tags present in the *Query Box* when a bookmark action occurs) ($\mu = 4.1,\ \sigma = 1.9$), which did not differ significantly from the average number of unique keywords selected per participant ($\mu = 4.9,\ \sigma = 2.29$).

Interestingly, although the event of the first bookmark in general aligns to the half-session cut and most control actions take place in the first part, it is possible to observe two different search patterns in the logged data. The streams in Figure 14(a) represent control (all four kinds) and bookmark actions for each user, such that the x-axis in a user diagram indicates the order of execution of logged actions (regardless of time offset) and the two levels in the y-axis split control (on top) from bookmark actions (bottom). These stream-based graphics reveal, on one hand, that eight participants (2, 6, 8, 11, 15, 17, 18, and 20) conducted a two-step search, namely: they performed several tag manipulations and added all the bookmarks only after they had finished refining the document ranking. According to Qu and Furnas [42], users following this type of search pattern tend to build internal topical constructs and only create external constructs (folders) and add documents to them once the structure is clear in their minds. On the other hand, the logs for eight other participants (3, 4, 5, 7, 9, 14, 16, and 19) denote an iterative pattern, as they performed series of ranking updates intertwined with bookmarking actions. This search pattern seems to align with the berry-picking cognitive model for information retrieval [5], whereby search evolves and occurs bit by bit; that is, a user constantly changes the search terms after evaluating the returned results. We can also detect a third group of four participants (1, 10, 12, and 13) that fall in a fuzzy area between the two-step and the iterative search strategies, as they performed all bookmarks almost uninterruptedly with only one ranking refinement in between.

We picked one representative case of each pattern type, namely, participant 15 (P15) for two-step search and participant 16 (P16) for the iterative strategy. Then we plotted their logged actions in more detail and contrasted them against the exploratory search model in Figure 1. The action paths in Figure 14(b) add an intermediate level illustrating inspection actions, that is, opening a document to see its full content. Both participants started their sessions with five control actions (ranking updates). These initial actions denote a trial-and-error exploratory phase. Thereafter, they opted for different strategies: P15 inspected dozens of documents before deciding to create the first bookmark, while P16 immediately started to save resources. What they both have in common is that the subsequent updates (control actions) and bookmarks are preceded and followed



(a) Search paths logged in Study II for 20 participants



(b) Search paths logged in Study II

Fig. 14. Search strategies in study II.

by several document inspections (between seven and 33), which sheds light on the need of users to seek explanations at every stage in the discovery process. We can also see that two kinds of situations triggered awareness in the participants, meaning that they realized their information needs had changed and hence it was time to update the document ranking. One case is after reading several documents (orange paths ascending from mid- to upper level in the y-axis), and the second case is when the user possibly discovered new pieces of information after bookmarking a document (orange paths from bottom to upper level). Overall, Figure 14 suggests that the cycle of *exploration-explanation-awareness* introduced in Figure 1 is applicable to either the two-step or the iterative search patterns, even though the internal processes that make the user aware of changes in information needs may vary.

Summarizing, we observed that users mostly tended to conduct the search process following either a two-step or an iterative pattern. Yet, regardless of the search strategy, the event of the first bookmark in the middle of an average session suggests that participants occupied half their time for sense making and only then were they ready to assess the relevance or usefulness of a document.

5.4 Usability Analysis

In addition to questions for specific UI features, participants also filled out a *Software Usability Scale* (SUS) questionnaire [8], a standard poststudy questionnaire for subjective assessment of usability. Participants were not native English speakers, and hence we chose a version with all positive-tone questions [50] for better understanding. To keep consistency with the scoring scale in the *uRank*-specific questions, we used a 7-point Likert scale instead of a 5-point one (in this case 1 = strongly disagree, 7 = strongly agree). User responses were multiplied by 1.66 instead of 2.5 to obtain overall SUS scores in a range between 0 and 100. Thus, the score s_i for question x_i was computed as $s_i = (x_i - 1) * 1.6$.

Averaging over all questions and participants, the mean raw score amounted to 81.8 ($\sigma = 13.9$). *uRank* falls in the 90th to 95th percentile range in the curved grading scale interpretation of SUS scores [50], and thus obtained an **A grade**. These scores are also subdivided into *Usable* (questions 1, 2, 3, 5, 6, 7, and 8) and *Learnable* (4 and 10) subscales [30]. Adjusting multipliers for a 7-point Likert scale (2.08 and 8.33, respectively), *uRank* scored an A for *Usable* ($\mu = 80.3$, $\sigma = 14.7$) and an A+ for *Learnable* ($\mu = 87.5$, $\sigma = 14.7$).

At the end of the study, users were asked to share their general impressions of the system. Two participants reported some delays in the ranking update after changing tag weights, and a few expressed that they would prefer softer colors for tags and bars or even less color diversity. Nonetheless, most of the positive answers agreed that overall usability and ease of use were good. A few answers even highlighted that animations made it easy to follow the effects of their actions.

5.5 Discussion and Limitations

This study revealed which parts of the tool proved most useful and easy to handle. More importantly, it provided some insights on how exploratory search unfolded: for example, control actions occurred mostly before the first bookmark, and then users dedicated their time to find more interesting documents. This was, to some extent, surprising, as we expected a decreasing yet less abrupt frequency of ranking updates.

Particularly, this study falls on the opposite side of study I. The task did not impose any goals or reasons to actually pursue a dedicated search. Therefore, perhaps people were only motivated to try the tool once and see how it works, but did not actually engage in exploring and learning. We expect that conscious exploration would involve several changes of interest and further interactions. The short duration of an average session (4’50”) corroborates this assumption.

We believe the only way to produce a realistic situation would be through a longitudinal study, for example, having an online service where people can obtain an actual profit after interacting with it. Another option could be a more balanced setup that is less controlled than study I and less free than study II. However, defining the optimal setup for this kind of evaluation is out of the scope of this article.

6 CONCLUSIONS AND FUTURE WORK

This article introduced a visual analytic tool for exploratory search of textual documents. The reasoning line for its visual and interactive design is strongly influenced by a conceptual model based on a cycle of awareness, exploration, and explanation. This work also provided details about some lessons learned and incremental changes implemented along the way.

Although originally planned as a recommending interface, the whole concept of *uRank* (UI plus analytic methods) can be extended to generic search interfaces for not only unstructured data like text but also structured data. Doing so would require enriching the input data with semantic information, such as tags or named entities.

We also presented two complementary user studies that either validated some aspects of the design or showed the path for further improvement. Examples of the latter might be a more intuitive mechanism for multiple drag-and-drop and the addition of an overview component to navigate large lists. The first study consisted of a highly controlled comparative evaluation against a traditional list-based UI. Results revealed that participants found it significantly more relaxing to work with *uRank*, despite no major differences in performance or speed. In fact, most of them reported their wish to start actively using *uRank* in their scientific endeavors (e.g., paper writing). The limitations of a strictly controlled study served as a starting point for the second study. In this case, we asked users to freely explore documents with *uRank* and bookmark the relevant ones within a topic of interest. This study shed light on interaction paths and browsing strategies followed during exploratory search. Moreover, the tool received positive critique, as supported by the poststudy questionnaires.

Future work includes extending the text-mining methods, for example, by enriching keywords with synonyms and semantic relationships. Ultimately, we plan to leverage bookmark logs collected during both studies and use them as feedback to improve recommendations with folksonomy-based information, closing the interactive loop with the RS, as shown in Figure 2.

APPENDIXES

A QUESTIONNAIRES

A.1 Questionnaire Used in Study I

Figure 15 summarizes the distribution of responses to the NASA TLX scale gathered during study I.

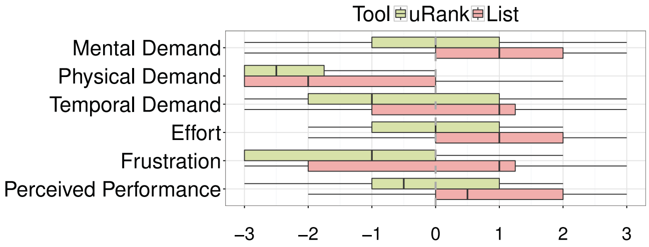


Fig. 15. Boxplots show distribution of subjective feedback provided in study I.

A.2 Questionnaire Used in Study II

Responses to UI-specific questions answered by participants in study II are presented in Figure 16.

B MULTILEVEL MODEL FOR REPEATED MEASURES DESIGN

MLMs allow one to model the fact that some observations come from the same entities and are hence correlated. Several authors report their benefits over repeated-measures ANOVA [43] and even provide step-by-step guidance with statistical software packages [11]. These models are extensively applied in behavioral studies but not yet quite adopted in HCI evaluations. Thus, this section has been added for clarification purposes.

Recall that all participants in study I performed four tasks under all combinations of the independent variables; hence, we assume that mean differences across sampling units (i.e., participants) create a constant dependency over the independent variables. The first level of our MLM is formally defined as a linear model in Equation (4), where $Y_{i,j}$ is the outcome variable (in our case

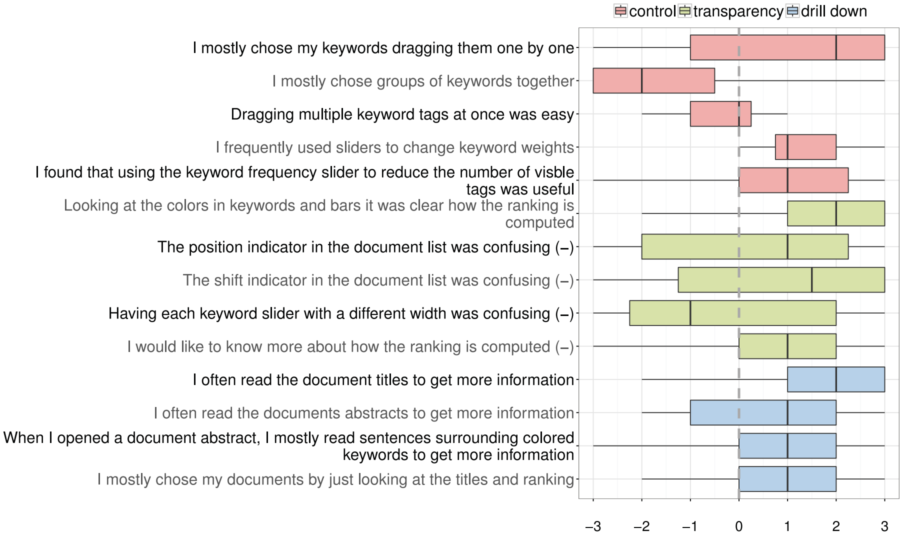


Fig. 16. Boxplots show distribution of subjective feedback provided in study II.

Table 4. Multilevel Mixed-Effects Model for Workload

Goodness of Fit	Model A		Model B	
$-2LL$	764		762	
AIC	-775		-779	
BIC	-791		-799	
Fixed Effects	Coefficients (SE)	p	Coefficients (SE)	p
Intercept (γ_{00} in b_{0j})	46.3 (2.92)	***	46.3 (2.95)	***
Tool (γ_{10} in b_1)	-7.7 (1.28)	***	-7.7 (1.30)	***
Items (γ_{20} in b_2)			0.7 (0.92)	
Tool × Items			-0.2 (0.92)	
Random Effects	Variance (SD)		Variance (SD)	
Level 1 Residual (ϵ_{ij})	13.47 (3.67)		13.40 (3.66)	
Level 2 Residual (u_{0j})	723.07 (26.89)		722.00 (26.87)	
Total Residuals	736.54		735.40	

Incremental Models: Model A only includes *tool* as predictor, whereas model B incorporates *#items* and the interaction thereof. The *Fixed Effects* part shows that model B did not improve model A; that is, *#items* and the interaction are not significantly good predictors for workload. The *Random Effects* part indicates the amount of variance explained by the lowest-level residuals (ϵ_{ij}) and the higher-level residual (u_{0j}).
Goodness of fit:–2log-likelihood ($-2LL$), Akaike's information criterion (AIC), and Schwarz's Bayesian criterion (BIC) are interpreted as the smaller the better.

either workload or completion time); $Tool_{i,j}$ and $Items_{i,j}$ are the levels assigned to the dependent variables *tool* and *#items* for participant i and task j, respectively; and $(Tools \times Items)_{i,j}$ is the interaction of the main effects. b_{0j} is a random intercept and ϵ_{ij} is the residual for each observation.

$$Y_{i,j} = b_{0j} + b_1\ Tool_{i,j} + b_2\ Items_{ij} + b_{12}\ (Tools \times Items)_{ij} + \epsilon_{ij} \tag{4}$$

The coefficient b_{0j} is referred to as random intercept because it adds a component to measure the variability in intercepts across participants. The second level of the model is thus defined in

Equation (5). Coefficients $b_1 = \gamma_{10}$ and $b_2 = \gamma_{20}$ are the average slopes for *Tool* and *Items* across all participants, such that the subindices in γ_{10} and γ_{20} indicate that they are coefficients of level 1.

$$b_{0j} = \gamma_{00} + u_{0j}, \tag{5}$$

where γ_{00} is the average intercept across all participants and u_{0j} is a level 2 residual, that is, for each participant. The two levels of our MLM are then combined in Equation (6):

$$Y_{i,j} = b_0 + b_1\ Tool_{i,j} + b_2\ Items_{i,j} + b_{12}\ (Tools \times Items)_{i,j} + (u_{0j} + \epsilon_{ij}). \tag{6}$$

Finally, Table 4 presents a version of our model with only *Tool* as a predictor, whereas model B incorporates *Items* and the interaction of the main effects. Looking at goodness-of-fit measures, it is evident that model A was not improved by these additions, given that the number of items had no significant effect on workload measures. The middle part of the table lists the coefficients obtained for the fixed effects and the bottom the residuals of the mixed part of the model. Note that adding a second-level residual allows one to explain 98% of the total variance.

REFERENCES

[1] Jae-wook Ahn and Peter Brusilovsky. 2013. Adaptive visualization for exploratory information retrieval. *Information Processing and Management* 49, 5 (2013), 1139–1164. DOI: http://dx.doi.org/10.1016/j.ipm.2013.01.007

[2] Jae-wook Ahn, Peter Brusilovsky, Daqing He, Jonathan Grady, and Qi Li. 2008. Personalized web exploration with task models. *Proceeding of the 17th International Conference on World Wide Web (WWW'08)*, 1–10. DOI: http://dx.doi.org/10.1145/1367497.1367499

[3] Keith Andrews, Wolfgang Kienreich, Vedran Sabol, Jutta Becker, Georg Droschl, Frank Kappe, Michael Granitzer, Peter Auer, and Klaus Tochtermann. 2002. The Infosky visual explorer: Exploiting hierarchical structure and document similarities. *Information Visualization* 1, 3/4 (Dec. 2002), 166–181. DOI: http://dx.doi.org/10.1057/palgrave.ivs.9500023

[4] Anne Aula. 2005. *Studying User Strategies and Characteristics for Developing Web Search Interfaces*. Ph.D. Dissertation. University of Tampere. https://tampub.uta.fi/handle/10024/67544

[5] Marcia J. Bates. 1989. The design of browsing and berrypicking techniques for the online search interface. *Online Review* 13, 5 (May 1989), 407–424. DOI: http://dx.doi.org/10.1108/eb024320

[6] David M. Blei and John D. Lafferty. 2009. Topic models. *Text Mining: Classification, Clustering, and Applications* 10, 71 (2009), 34. DOI: http://dx.doi.org/10.1145/1143844.1143859

[7] Eric Brill. 1992. A simple rule-based part of speech tagger. In *Proceedings of the 3rd Conference on Applied Natural Language Processing (ANLC'92)*. Association for Computational Linguistics, Stroudsburg, PA, 152–155. DOI: http://dx.doi.org/10.3115/974499.974526

[8] John Brooke. 1996. SUS - A quick and dirty usability scale. *Usability Evaluation in Industry* 189, 194 (1996), 4–7. DOI: http://dx.doi.org/10.1002/hbm.20701

[9] Andy Cockburn, Amy Karlson, and Benjamin B. Bederson. 2009. A review of overview+detail, zooming, and focus+context interfaces. *ACM Computing Surveys* 41, 1, Article 2 (Jan. 2009), 31 pages. DOI: http://dx.doi.org/10.1145/1456650.1456652

[10] Douglass R. Cutting, David R. Karger, Jan O. Pedersen, and John W. Tukey. 1992. Scatter/gather: A cluster-based approach to browsing large document collections. In *Proceedings of the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'92)*. ACM, New York, NY, 318–329. DOI: http://dx.doi.org/10.1145/133160.133214

[11] Andy Field, Jeremy Miles, and Zoë Field. 2012. *Discovering Statistics Using R*. Sage, London.

[12] Debasis Ganguly, Manisha Ganguly, Johannes Leveling, and Gareth J. F. Jones. 2013. TopicVis: A GUI for topic-based feedback and navigation. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'13)*. ACM, New York, NY, 1103–1104. DOI: http://dx.doi.org/10.1145/2484028.2484202

[13] Dorota Glowacka, Tuukka Ruotsalo, Ksenia Konuyshkova, Kumaripaba Athukorala, Samuel Kaski, and Giulio Jacucci. 2013. Directing exploratory search: Reinforcement learning from user interactions with keywords. In *Proceedings of the 2013 International Conference on Intelligent User Interfaces (IUI'13)*. 117–128. DOI: http://dx.doi.org/10.1145/2449396.2449413

[14] Erick Gomez-Nieto, Frizzi San Roman, Paulo Pagliosa, Wallace Casaca, Elias S. Helou, Maria Cristina F. de Oliveira, and Luis Gustavo Nonato. 2014. Similarity preserving snippet-based visualization of web search results. *IEEE Transactions on Visualization and Computer Graphics* 20, 3 (March 2014), 457–470. DOI: http://dx.doi.org/10.1109/TVCG.2013.242

[15] Samuel Gratzl, Alexander Lex, Nils Gehlenborg, Hanspeter Pfister, and Marc Streit. 2013. LineUp: Visual analysis of multi-attribute rankings. *IEEE Transactions on Visualization and Computer Graphics* 19, 12 (Dec. 2013), 2277–2286. DOI:http://dx.doi.org/10.1109/TVCG.2013.173

[16] Brynjar Gretarsson, John O'Donovan, Svetlin Bostandjiev, Tobias Höllerer, Arthur Asuncion, David Newman, and Padhraic Smyth. 2012. TopicNets: Visual analysis of large text corpora with topic modeling. *ACM Transactions on Intelligent Systems and Technology* 3, 2, Article 23 (Feb. 2012), 26 pages. DOI:http://dx.doi.org/10.1145/2089094.2089099

[17] Mark Harrower and Cynthia A. Brewer. 2003. ColorBrewer.org: An online tool for selecting colour schemes for maps. *Cartographic Journal* 40, 1 (June 2003), 27–37. DOI:http://dx.doi.org/10.1179/000870403235002042

[18] Marti A. Hearst. 1995. TileBars: Visualization of term distribution information in full text information access. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI'95).* ACM Press/Addison-Wesley Publishing Co., New York, 59–66. DOI:http://dx.doi.org/10.1145/223904.223912

[19] Marti A. Hearst. 2009. *Search User Interfaces.* Cambridge University Press, New York.

[20] Jeffrey Heer and George Robertson. 2007. Animated transitions in statistical data graphics. *IEEE Transactions on Visualization and Computer Graphics* 13, 6 (Nov. 2007), 1240–1247. DOI:http://dx.doi.org/10.1109/TVCG.2007.70539

[21] Jonathan L. Herlocker, Joseph A. Konstan, and John Riedl. 2000. Explaining collaborative filtering recommendations. In *Proceedings of the 2000 ACM Conference on Computer Supported Cooperative Work.* 241–250. DOI:http://dx.doi.org/10.1145/358916.358995 arxiv:48

[22] Orland Hoeber and Xue Dong Yang. 2006. The visual exploration of web search results using hotmap. In *Proceedings of the Conference on Information Visualization (IV'06).* IEEE Computer Society, Washington, DC, 157–165. DOI:http://dx.doi.org/10.1109/IV.2006.108

[23] Yuening Hu, Jordan Boyd-Graber, and Brianna Satinoff. 2011. Interactive topic modeling. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1 (HLT'11).* Association for Computational Linguistics, Stroudsburg, PA, 248–257. http://dl.acm.org/citation.cfm?id=2002472.2002505

[24] Laurent Itti, Christof Koch, and Ernst Niebur. 1998. A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 20, 11 (1998), 1254–1259. http://portal.acm.org/citation.cfm?id=297870

[25] Antti Kangasrääsiö, Dorota Glowacka, and Samuel Kaski. 2015. Improving controllability and predictability of interactive recommendation interfaces for exploratory search. In *Proceedings of the 20th International Conference on Intelligent User Interfaces (IUI'15).* ACM, New York, 247–251. DOI:http://dx.doi.org/10.1145/2678025.2701371

[26] Judy Kay. 2006. Scrutable adaptation: Because we can and must. In *Proceedings of the 4th International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH'06).* Springer-Verlag, Berlin, 11–19. DOI:http://dx.doi.org/10.1007/11768012_2

[27] Roman Kern, Kris Jack, and Michael Granitzer. 2014. Recommending Scientific Literature: Comparing Use-Cases and Algorithms. Retrieved from http://arxiv.org/abs/1409.1357.

[28] Bart P. Knijnenburg, Svetlin Bostandjiev, John O'Donovan, and Alfred Kobsa. 2012. Inspectability and control in social recommenders. In *Proceedings of the 6th ACM Conference on Recommender Systems (RecSys'12).* ACM, New York, 43–50. DOI:http://dx.doi.org/10.1145/2365952.2365966

[29] Manoj Krishnan, Shawn Bohn, Wendy Cowley, Vern Crow, and Jarek Nieplocha. 2007. Scalable visual analytics of massive textual datasets. In *2007 IEEE International Parallel and Distributed Processing Symposium.* IEEE, 1–10. DOI:http://dx.doi.org/10.1109/IPDPS.2007.370232

[30] James R. Lewis and Jeff Sauro. 2009. The factor structure of the system usability scale. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 5619 LNCS (2009), 94–103. DOI:http://dx.doi.org/10.1007/978-3-642-02806-9_12

[31] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval.* Cambridge University Press, New York.

[32] Gary Marchionini. 2006. Exploratory search: From finding to understanding. *Communications of the ACM* 49, 4 (April 2006), 41–46. DOI:http://dx.doi.org/10.1145/1121949.1121979

[33] Tien Nguyen and Jun Zhang. 2006. A novel visualization model for web search results. *IEEE Transactions on Visualization and Computer Graphics* 12, 5 (Sept. 2006), 981–988. DOI:http://dx.doi.org/10.1109/TVCG.2006.111

[34] Mark Nolan. 2008. IA column: Exploring exploratory search. *Bulletin of the American Society for Information Science and Technology* 34, 4 (2008), 38–41. DOI:http://dx.doi.org/10.1002/bult.2008.1720340410

[35] John O'Donovan, Barry Smyth, Brynjar Gretarsson, Svetlin Bostandjiev, and Tobias Höllerer. 2008. PeerChooser: Visual interactive recommendation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI'08).* ACM, New York, 1085–1088. DOI:http://dx.doi.org/10.1145/1357054.1357222

[36] Kai A. Olsen, Robert R. Korfhage, Kenneth M. Sochats, Michael B. Spring, and James G. Williams. 1993. Visualization of a document collection: The vibe system. *Information Processing and Management* 29, 1 (1993), 69–81. DOI:http://dx.doi.org/10.1016/0306-4573(93)90024-8

[37] Denis Parra, Peter Brusilovsky, and Christoph Trattner. 2014. See what you want to see: Visual user-driven approach for hybrid recommendation. In *Proceedings of the 19th International Conference on Intelligent User Interfaces (IUI'14)*. ACM, New York, 235–240. DOI: http://dx.doi.org/10.1145/2557500.2557542

[38] Peter Pirolli. 2007. Cognitive models of human information interaction. In *Handbook of Applied Cognition* (2nd ed.), Francis Durso (Ed.). Wiley & Sons, New York.

[39] Daniel M. Russell, Mark J. Stefik, Peter Pirolli, and Stuart K. Card. 1993. The Cost Structure of Sensemaking. In *Proceedings of the INTERACT'93 and CHI'93 Conference on Human Factors in Computing Systems (CHI'93)*. ACM, New York, 269–276. DOI: 10.1145/169059.169209

[40] Peter L. T. Pirolli. 2007. *Information Foraging Theory: Adaptive Interaction with Information*. Oxford University Press,, New York.

[41] M. F. Porter. 1980. An algorithm for suffix stripping. *Program: Electronic Library and Information Systems* 14, 3 (1980), 130–137. DOI: http://dx.doi.org/10.1108/eb046814

[42] Yan Qu and George W. Furnas. 2008. Model-driven formative evaluation of exploratory search: A study under a sensemaking framework. *Information Processing and Management* 44, 2 (2008), 534–555. DOI: http://dx.doi.org/10.1016/j.ipm.2007.09.006

[43] Hugo Quené and Huub Van Den Bergh. 2004. On multi-level modeling of data from repeated measures designs: A tutorial. *Speech Communication* 43, 1–2 (2004), 103–121. DOI: http://dx.doi.org/10.1016/j.specom.2004.02.004

[44] Ramana Rao and Stuart K. Card. 1994. The table lens: Merging graphical and symbolic representations in an interactive focus + context visualization for tabular information. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI'94)*. ACM, New York, 318–322. DOI: http://dx.doi.org/10.1145/191666.191776

[45] Manuela Rauch, Werner Klieber, Ralph Wozelka, Santokh Singh, and Vedran Sabol. 2015. Knowminer search - A multi-visualisation collaborative approach to search result analysis. In *2015 19th International Conference on Information Visualisation (iV'15)*. 379–385. DOI: http://dx.doi.org/10.1109/iV.2015.72

[46] Harald Reiterer, Gabriela Tullius, and T. M. Mann. 2005. Insyder: A content-based visual-information-seeking system for the web. *International Journal on Digital Libraries* 5, 1 (2005), 25–41. DOI: http://dx.doi.org/10.1007/s00799-004-0111-y

[47] Francesco Ricci, Lior Rokach, Bracha Shapira, and Paul B. Kantor. 2010. *Recommender Systems Handbook*. Springer-Verlag New York, New York.

[48] Tuukka Ruotsalo, Giulio Jacucci, Petri Myllymäki, and Samuel Kaski. 2015. Interactive intent modeling: Information discovery beyond search. *Communications of the ACM* 58, 1 (2015), 86–92. DOI: http://dx.doi.org/10.1145/2656334

[49] Tuukka Ruotsalo, Jaakko Peltonen, Manuel Eugster, Dorota Glowacka, Ksenia Konyushkova, Kumaripaba Athukorala, Ilkka Kosunen, Aki Reijonen, Petri Myllymäki, Giulio Jacucci, and Samuel Kaski. 2013. Directing exploratory search with interactive intent modeling. In *Proceedings of the 22nd ACM International Conference on Conference on Information and Knowledge Management*. 1759–1764. DOI: http://dx.doi.org/2505644

[50] Jeff Sauro and James R. Lewis. 2012. *Quantifying the User Experience*. Morgan Kaufmann.

[51] Christin Seifert, Johannes Jurgovsky, and Michael Granitzer. 2014. FacetScape: A visualization for exploring the search space. In *Proceedings 18th International Conference on Information Visualization*. 94–101. DOI: http://dx.doi.org/10.1109/IV.2014.49

[52] Christin Seifert, Barbara Kump, Wolfgang Kienreich, Gisela Granitzer, and Michael Granitzer. 2008. On the beauty and usability of tag clouds. In *Proceedings of the 2008 12th International Conference Information Visualisation (IV'08)*. IEEE Computer Society, 17–25. DOI: http://dx.doi.org/10.1109/IV.2008.89

[53] Guy Shani and Noam Tractinsky. 2013. Displaying relevance scores for search results. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'13)*. ACM, New York, 901–904. DOI: http://dx.doi.org/10.1145/2484028.2484112

[54] Ben Shneiderman, Donald Byrd, and W. Bruce Croft. 1998. Sorting out searching: A user-interface framework for text searches. *Communications of the ACM* 41, 4 (April 1998), 95–98. DOI: http://dx.doi.org/10.1145/273035.273069

[55] Anselm Spoerri. 2004. Coordinated views and tight coupling to support meta searching. In *Proceedings of the 2nd International Conference on Coordinated & Multiple Views in Exploratory Visualization (CMV'04)*. IEEE Computer Society, 39–48. DOI: http://dx.doi.org/10.1109/CMV.2004.5

[56] Marc Streit and Nils Gehlenborg. 2014. Bar charts and box plots. *Nature Methods* 11, 2 (Feb. 2014), 117. DOI: http://dx.doi.org/10.1038/nmeth.2807

[57] Alistair Sutcliffe and Mark Ennis. 1998. Towards a cognitive theory of information retrieval. *Interacting with Computers* 10, 3 (1998), 321–351. DOI: http://dx.doi.org/10.1016/S0953-5438(98)00013-7 {HCI} and Information Retrieval.

[58] K. Swearingen and R. Sinha. 2001. Beyond algorithms: An HCI perspective on recommender systems. In *Proceedings of the ACM SIGIR 2001 Workshop on Recommender Systems*. ACM Press, 1–11.

[59] Nava Tintarev and Judith Masthoff. 2012. Evaluating the effectiveness of explanations for recommender systems. *User Modeling and User-Adapted Interaction* 22, 4–5 (Oct. 2012), 399–439. DOI: http://dx.doi.org/10.1007/s11257-011-9117-5

[60] Katrien Verbert, Denis Parra, Peter Brusilovsky, and Erik Duval. 2013. Visualizing recommendations to support exploration, transparency and controllability. *Proceedings of the 2013 International Conference on Intelligent User Interfaces (IUI'13)*, 351. DOI: http://dx.doi.org/10.1145/2449396.2449442

[61] Michelle Q. Wang Baldonado, Allison Woodruff, and Allan Kuchinsky. 2000. Guidelines for using multiple views in information visualization. In *Proceedings of the Working Conference on Advanced Visual Interfaces (AVI'00)*. ACM, New York, 110–119. DOI: http://dx.doi.org/10.1145/345513.345271

[62] Matthew Ward, Georges Grinstein, and Daniel Keim. 2010. *Interactive Data Visualization: Foundations, Techniques, and Applications*. A. K. Peters, Natick, MA.

[63] Colin Ware. 2012. *Information Visualization: Perception for Design* (3rd ed.). Morgan Kaufmann Publishers, San Francisco.

[64] Ka-Ping Yee, Kirsten Swearingen, Kevin Li, and Marti Hearst. 2003. Faceted metadata for image search and browsing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI'03)*. ACM, New York, 401–408. DOI: http://dx.doi.org/10.1145/642611.642681

Received July 2016; revised November 2016; accepted February 2017