Rank As You Go: User-Driven Exploration of Search Results

Cecilia di Sciascio
Know-Center GmbH
Graz, Austria
cdisciascio@know-center.at

Vedran Sabol
Know-Center GmbH
Graz, Austria
vsabol@know-center.at

Eduardo E. Veas
National University of Cuyo
Mendoza, Argentina
eveas@know-center.at

ABSTRACT
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 *uRank* and investigate interactive methods for understanding, refining and reorganizing documents on-the-fly as information needs evolve. *uRank* includes views summarizing the contents of a recommendation set and interactive methods conveying the role of users' interests through a recommendation ranking. A formal evaluation showed that gathering items relevant to a particular topic of interest with *uRank* incurs in lower cognitive load compared to a traditional ranked list. A second study consisting in an ecological validation reports on usage patterns and usability of the various interaction techniques within a free, more natural setting.

ACM Classification Keywords
H.5.2. Information Interfaces and Presentation: User Interfaces – Graphical user interfaces (GUI), Interaction styles (e.g., commands, menus, forms, direct manipulation), Evaluation/methodology;; H.3.3. Information Storage and Retrieval: Information Search and Retrieval – Information filtering, Search process; I.2.7. Artificial Intelligence: Natural Language Processing – Text analysis

Author Keywords
search interface; interaction; visual analytics; user study

INTRODUCTION
With the advent of electronic archival, seeking for information occupies a large portion of our daily productive time. Thus, the skill to find and organize the right information has become paramount. Exploratory search is 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. For example, after inspecting a few documents related to "robots", sub-topics like "human-robot interaction" or "virtual environments" could attract the user's attention. Exploration requires careful inspection of at least a few titles and abstracts, when not full documents, before becoming familiar with the underlying topic.

Advanced search engines and recommender systems (RS) have grown as the preferred solution for contextualized search by narrowing down the number of entries that need to be explored at a time. However, traditional information retrieval (IR) systems strongly depend on precise user-generated queries that should be iteratively reformulated to express evolving information needs. Formulating queries has proven to be more complicated for humans than plainly recognizing information in a visual manner [12]. Hence, the combination of IR with machine learning and HCI techniques has led to a shift towards – mostly Web-based – browsing search strategies that rely on on-the-fly selections, navigation and trial-and-error [23]. As users manipulate data through visual elements, they are able to drill down and find patterns, relations or different levels of detail that would otherwise remain invisible to the bare eye [43]. Moreover, well-designed interactive interfaces can effectively address information overload issues that may arise due to limited attention span and human capacity to absorb information at once.

In turn, RS can be more limited than IR systems if they do not tackle trust factors that hinder user engagement in exploration. As Swearingen et al. [38] pointed out in their seminal work, the RS has to persuade the user to try the recommended items. To fulfill such challenge not only the recommendation algorithm has to fetch items effectively, but also the user interfaces must deliver recommendations in a way that they can be compared and explained [31]. Explanatory interfaces increase confidence in the system (*trust*) by explaining how the system works (*transparency*) [39] and allowing users to tell the system when it is wrong (*scrutability*) [18]. Hence, to warrant increased user involvement the RS has to justify recommendations and let the user customize their generation.

In this work we focus mainly on transparency, controllability and, to some extent, on predictability features that support: (a) exploration of textual document recommendations and (b) refinement of evolving information needs. *uRank* is a visual analytics approach that automatically generates an interactive keyword-based overview of the document collection. It allows users to discover keyword-document relationships – query preview to predict the effect of keyword selection –, as well

IUI 2016, March 7–10, 2016, Sonoma, CA, USA.

http://dx.doi.org/10.1145/2856767.2856797

as keyword-keyword relationships – possible key phrases –. Users 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 that took place in both, a controlled and an unstructured context.

Our contributions are summarized as: *(i)* a highly controllable and transparent UI for exploratory search of textual documents and *(ii)* two user studies covering the benefits of *uRank* with respect to a traditional list-based UI and usability aspects.

BACKGROUND

Search Result Visualization

Modern search interfaces assist user exploration in a variety of ways. For example, query expansion techniques like *Insyder*'s Visual Query [30] address the query formulation problem by leveraging stored related concepts to help the user extend the initial query. Tile-based visualizations like *TileBars* [13] and *HotMap* [15] make an efficient use of space to convey relative frequency of query terms through – gray or color – shaded squares, and in the case of the former, also their distribution within documents and relative document length. This paradigm aims to foster analytical understanding of Boolean-type queries, 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 [44, 33]. More recently, interfaces supporting rather natural facet-type visual filtering, e.g. geographic or temporal, have also been proposed [29]. However, faceted search relies on structured information, i.e. metadata categories, thus it hardly supports topic-wise exploration of unstructured data.

Rankings conveying document relevance have been discouraged as opaque an under-informative [13]. 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 [35] suggests that: *i)* users prefer bars over numbers or the absence of graphical explanations of relevance scores, and *ii)* 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, thus they are mostly apt for sets no larger than a few tens of documents. Focus+Context and Overview+Detail techniques [28, 15] sometimes help overcome this limitation while alternative layouts like *RankSpiral*'s [36] rolled list can scale up to hundreds and maybe thousands of documents. Other approaches such as *WebSearchViz* [24] and *ProjSnippet* [8] propose complementary visualizations to ordered lists, yet unintuitive context switching is a potential problem when analyzing different aspects of the same document.

Although ranked list are not a novelty, our approach attempts to leverage the advantages provided by lists; i.e. user familiarity, and augment them with stacked-bar charts to convey document relevance and query term contribution in a transparent manner. *Insyder*'s bar graph [30] 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 [37]. Additionally, we rely on interaction as the key to provide controllability over the ranking criteria and hence support browsing-based exploratory search.

LineUp [9] has proven the simplicity and usefulness of stacked bars to represent multi-attribute rankings. Despite targeting data of different nature – *uRanks*'s domain is rather unstructured with no measurable attributes –, the visual technique itself served as inspiration for our work.

Recommending Interfaces

In recent years, considerable efforts have been invested into leveraging the power of social RS through visual interfaces [25, 19]. As for textual content, *TalkExplorer* [40] and *SetFusion* [26] 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. [17] highlighted that not only allowing the user to influence the RS is important, but also 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.

Topic Analysis

In addition to methods for visualizing item relevance or distribution of query terms along search results, we consider 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 [34]. 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 [5] are able to handle very large pools by building hierarchical structures for top-down exploration. IN-SPIRE [20] and InfoSky [1] provide cluster browsing interfaces based on spatial metaphors: a landscape and the outer-space, respectively. Hierarchical cluster exploration is not a trivial task, therefore they are rarely adopted by real-world systems.

Another alternative is topic models, which performs a generative approach, e.g. latent Dirichlet allocation, to capture themes inherent to a document collection [2]. Typical UIs present a topic overview of a collection and allow for further exploration at multiple levels via zooming with keyword search [7] or by navigating a network of interconnected documents, like in *TopicNets* [10]. 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 basing on term co-occurrence. This flexibility could also turn into a weakness, as the topics generated are often not interesting or relevant to users [16]. 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 [16]. 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.

URANK VISUAL ANALYTICS

uRank is a visual analytics approach that combines lightweight text analytics and an augmented ranked list to assist in exploratory search of textual documents. Figure 1 depicts the workflow between automatic and interactive mechanisms. 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, i.e. titles and abstracts. The Web-based implementation is currently fed by a RS connected to several sources.
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 interactively select them 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.
7. The user can select a single document to access more detailed information.
8. Once the user finds a document that suits their search interest, they can add it to their own collection.

User-driven actions (4, 5, 7 and 8) highly depend on the user's search strategy, thus they are rather iterative and interchangeable.

The User Interface

uRank's UI layout is arranged in a multiview fashion that displays different levels of abstraction of a document collection:

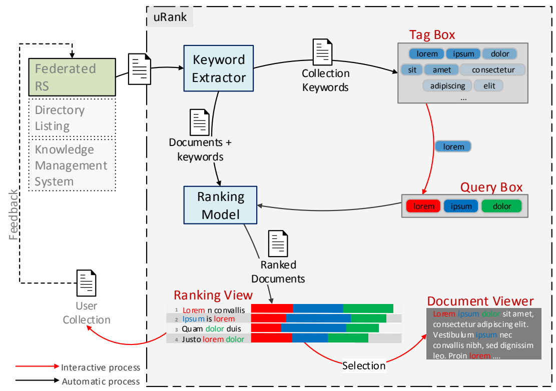


Figure 1. *uRank* visual analytics workflow showing automatic (black arrows) and interactive mechanisms (red arrows)

Collection overview. The *Tag Box* (Figure 2.A) summarizes the entire collection through augmented keyword tags.

Documents overview. The *Document List* shows titles along with ranking information and the *Ranking View* displays stacked bar charts depicting document relevance scores (Figure 2.C and 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 2.B).

Document detailed view. For a document selected in the list, the *Document Viewer* (see section Details on Demand) displays the title and snippet with color-augmented keywords.

Display space requirements constrain the number of views and their space at a given time. At first, all views appeared juxtaposed, avoiding multiple overlapping views. In the latest version we added the *Bookmark Overview* (Figure 2.E) and removed the *Document Viewer* from the main view and is shown as a modal when the user requires deeper information about a particular document.

Interactions and Visual Design

Our approach relies on a suitable visual encoding and on interactive mechanisms. 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 [42], uncovering pieces of information in the data space that would otherwise pass unnoticed.

Exploring a Document Collection

The *Tag Box* provides a summary of the textual documents as a whole by presenting keywords as tags. Summarizing the collection in a few representative terms allows the user to scan the recommendations and grasp the general topic at a glance, before even reading any of them. This is particularly important in the context of collections brought by RS, where the user is normally not directly generating the queries that feed the search engine.

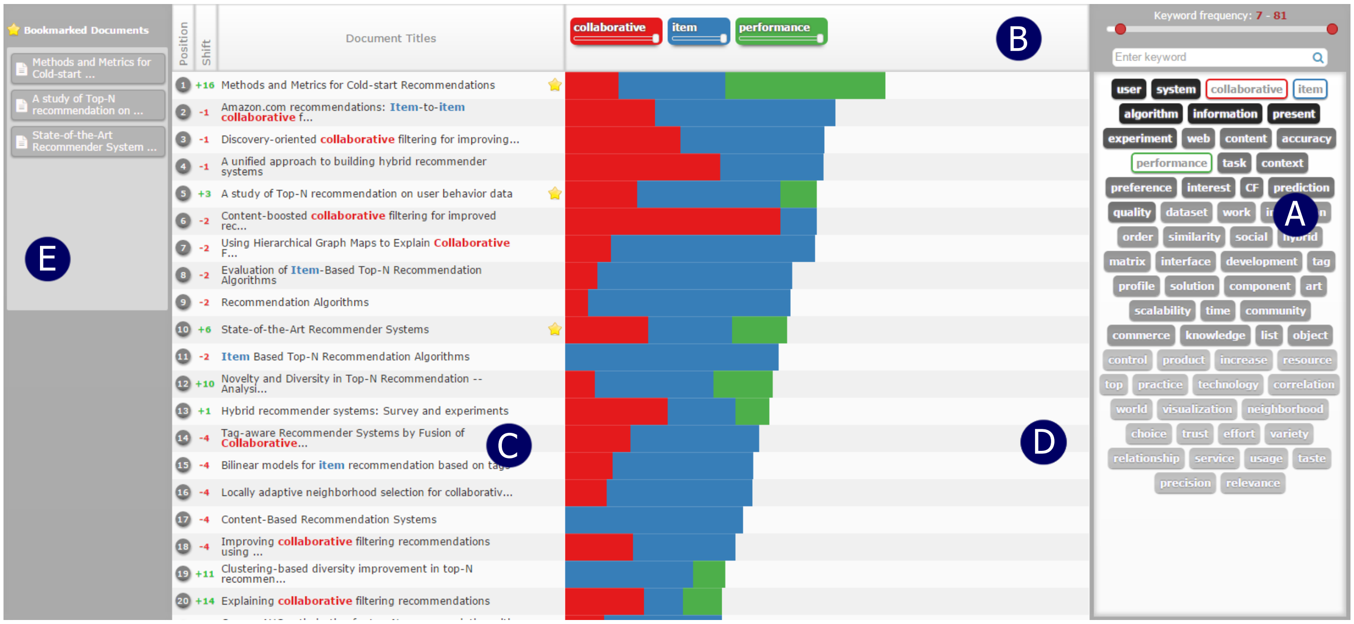


Figure 2. ***uRank*** **User Interface displaying documents related to** ***recommender systems*****, with ranking updated to match the keywords "collaborative", "item" and "performance". A. The** ***Tag Box*** **presents a keyword-based summary of the document collection, B. the** ***Query Box*** **contains keywords selected by the user, C. the** ***Document List*** **and D. the** ***Ranking View*** **present a list with augmented document titles and stacked bars indicating relevance scores, and E. the** ***Bookmark Overview*** **shows bookmarked documents.**

Tags are organized in descending order to convey document frequency (DF) and visually grouped in five gray-shaded clusters that represent keywords with similar frequencies. Redundant frequency coding is intentional and aims at maximizing distinctiveness among items in the keyword set [43].

At first glance, the *Tag Box* gives an outline of the covered topic in terms of keywords and their relative frequencies. Nevertheless, a bag-of-words representation alone does not supply further details about how a keyword relates to other keywords or documents. To bridge this gap, tags are augmented with two compact visual hints: *i)* the *co-occurence hint* appears as a red circle showing the number of frequently co-occurring keywords across all documents, and *ii)* the *document hint* consists in a pie chart that conveys the proportion of documents in which the keyword is contained. Both hints become visible on mouse over, along with a tooltip on top of the *Tag Box* that provides an accurate explanation of the hints' meaning (Figure 3(a)). Further inspection is possible by clicking on a particular tag, which has a twofold effect:

1. Unrelated documents are dimmed in the *Document List* and *Ranking View*, so that documents containing the keyword remain in focus – even if they are not currently ranked –. This feature allows for predicting the effect of selecting a keyword (Figure 4).

2. Co-occurring terms are brought to focus by dimming unrelated tags in the background (Figure 3(b)), in order to support the user in discovering possible key phrases within the collection.

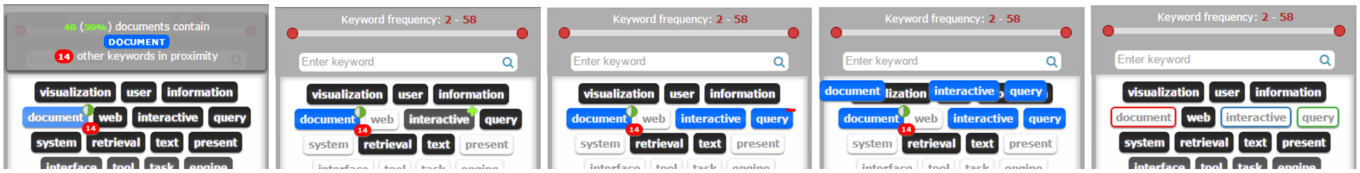
In the first-generation implementation, the user had to click on the *document-* and *co-occurrence hint* to trigger effects 1 and 2, respectively. A preliminary study showed that clicking on the small hints was unintuitive, therefore we simplified this interaction by triggering both effects together on tag clicks.

The study also revealed that users had difficulties finding a particular keyword, especially when the *Tag Box* was overly populated. For that reason, a text input field and a frequency range slider were added on top of the *Tag Box*. The former provides keyword search functionality, such that when the term is found, the corresponding tag is highlighted and the *Tag Box* scrolls to its position, or an error message pops up otherwise (Figure 5). 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 2.A the slider has been set to the range [7, 81], significantly reducing the number of tags in display.

Ranking Documents On The Fly

In theory, recommender and information retrieval systems produce lists where items are already sorted by their 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 criteria or clear rationale as to what makes and item more relevant than another. Hence, with *uRank* we address controllability and transparency 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.

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 manipula-



(a) (b) (c) (d) (e)

Figure 3. *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

tions in the *Query Box* (Figure 2.B): addition, weight change and deletion.

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 either dropped 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 3(c)) and drag them altogether (Figure 3(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 [11] –. 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.

In an earlier version, dragged tags were removed from the *Tag Box*, causing position adjustments in subsequent tags to fill the empty gaps. A previous evaluation revealed that this had the undesired effect that previously 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 in the *Tag Box* and the border is highlighted with the same categorical color assigned in the *Query Box*.

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.

Tag manipulations are forwarded to the *Ranking Model* as ranking parameters, which in turn feeds the ranking visualization. This visualization consists of a list of document titles (Figure 2.C) and stacked bar charts (Figure 2.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 [14]. In *uRank*, soft animated transitions for ranking-state changes and document selection help the user intuitively switch contexts. As Baldonado et al. [41] state in their rule of attention management, perceptual techniques lead the user's attention to the right view at the right time.

As 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. Stacked bars then appear *Ranking View*, 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.

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

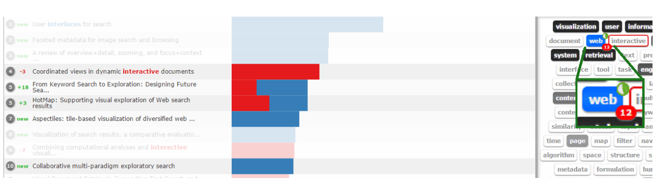


Figure 4. Document hints show which documents contain certain keyword, helping to predict the effect of selecting it

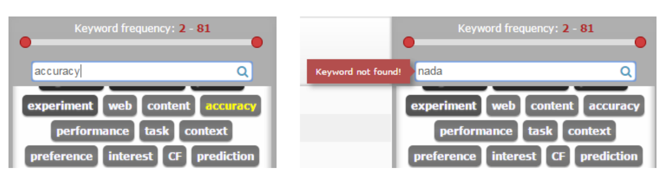


Figure 5. Keyword search in *Tag Box*. (left) Found keyword appears highlighted for a few seconds. (right) Error message pops up otherwise.

Figure 6. (*top*) Keyword tag before being dropped in *Tag Box*. (*bottom*) Dropped tag extended with weight slider and delete button. Background colors match a categorical scale and weights have been tuned.

immediate previous position. Position is denoted inside a gray circle, whereas shift is indicated as a color-coded number.

This visualization attempts to attract user's attention to likely relevant documents by bringing highly ranked ones to the top and pushing the rest to the bottom. In a previous generation, items with null score were hidden and the list height was shrunk or enlarged to fit only ranked items. Some users argued that such behavior was undesirable or confusing. Therefore, unranked items are currently grouped below the ranked ones, so that they remain accessible even though they are likely irrelevant for the given query.

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 list animations.

Details on Demand

Once the user identifies documents that seem worth further inspecting, he/she can drill down one by one to determine whether the initial assumption holds. The *Document Viewer*, as shown in (Figure 7 gives access to textual content – title and snippet – and available metadata for a particular document. Query terms are 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 pre-attentively recognize keywords in the text and perceive their general context prior to conscious reading.

Keyword Extraction

The aforementioned interactive features are supported by a combination of well-known text-mining techniques that extend the recommended 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

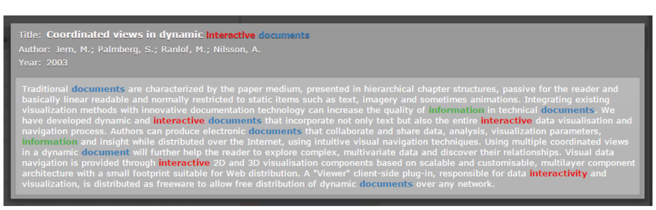


Figure 7. *Document Viewer* shows augmented title and abstract for a selected document. Color-coded terms match the tags in *Query Box*

stop words to remove meaningless terms. Firstly, we perform a part-of-speech tagging (POS tagging) [3] step to identify words that meet our criteria, i.e. common and proper nouns and adjectives. Filtering out non-frequent adjectives requires an extra step. Then, plural nouns are singularized, proper nouns are kept capitalized and terms in upper case, e.g. "IT", remain unchanged. We apply the Porter Stemmer method [27] 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 is thus conformed by stemmed versions of content-bearing terms.

Next, we generate a weighing scheme by computing TF-IDF (term frequency – inverse document frequency) for each term in a document vector. The 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), i.e. 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 certain threshold (by default 5) are taken into account. Note that terms used to label keyword tags are actual words and not plain stems. Scanning a summary of stemmed words would turn unintuitive for users. Thus, in order to allow for reverse stemming, we keep a record of all variations matching each stem, and pick the representative word as follows:
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 lower case and the other in upper case or capitalized, use it in lower case,
3. use a term that ends in 'ion', 'ment', 'ism' or 'ty',
4. use a term matching the stem,
5. use the shortest term.

To feed document hints, *uRank* attaches a list of bearing documents to each global keyword. For co-occurrence hints (Figure 2.A), *uRank* tracks by default keyword co-occurrences with a maximum word distance of 2 and at least 10 repetitions.

Document Ranking Computation

Quick content exploration in *uRank* depends on its ability to readily re-sort documents according to changing information needs. As the user manipulates keyword tags and builds queries from a subset of the global keyword collection, *uRank* computes documents scores to arrange them accordingly in a document ranking. We assume that some keywords are more important to the topic model than others and allow the user to assign weights to them.

Document scores are relevance measures for documents with respect to a query. As titles and snippets are the only content available for retrieved document surrogates, these scores are computed 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 document ranking.

The *Ranking Model* implements a vector space model to compute document-query similarity using the document vectors previously generated during keyword extraction. Nonetheless, a single relevance measure like cosine similarity alone is not enough to convey query-term contribution, given that the best overall matches are not necessarily the ones in which most query terms are found [13, 22]. 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. 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.

Given a document collection D and a set of weighted query terms T, such that $\forall t \in T : 0 \leq w_t \leq 1$; the relevance score for term t in document vector $d \in D$ for query terms T is calculated as follows:

$$s(t_d) = \frac{tfidf(t_d) \times w_t}{||d|| \times ||T||},$$

where $tfidf(t_d)$ is the tf-idf score for term t in document d and $||d||$ is the Euclidean norm for vector d. Note that every term t in query T is 1 , such that $1/||T||$ represents a single dot in its unit query vector. Thus the Euclidean norm of T equals the square root of the vector length, i.e. $||T|| = \sqrt{|T|}$.

The overall score of a document $S(d)$ is then computed as the sum of each individual term score $s(t_d)$. Finally, the collection D is sorted in descending order by overall score with the quicksort algorithm and ranking positions are assigned.

Implementation

uRank is a Web-based tool mainly implemented in JavaScript. We made use of libraries like *jQuery*[1] and *d3*[2] for the UI. Keyword extraction is performed entirely on the client side (due to project requirements). For POS-tagging, tokenization, stemming and tf-idf computation we leveraged *jspos*[3] and *NaturalJS* [4]. Other support libraries include *colorbrewer*, for color schemes, and Underscore.js, for diverse functionalities.

STUDY I: URANK VS BASELINE 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. The study followed a 2x2 repeated measures design with two independent variables: *tool*: uRank (*U*) and a baseline list-based UI (*L*) – with usual browser tools like Control+F search – and *#items*: *30* and *60*. Thus, every participant worked under the four possible combinations: **U-30**, **U-60**, **L-30** and **L-60**.

[1] https://jquery.com/
[2] http://d3js.org/
[3] https://code.google.com/p/jspos/
[4] https://github.com/amitamb/NaturalJS

Table 1. Cosine similarities between collections gathered during Study I

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

To counterbalance learning effects, we created data sets for 4 different topics and treated *topic* as random variable.

We recruited 24 participants (11 female, 13 male, between 22 and 37 years old), who were mainly graduate and post-graduate students from the computer science the medical domains. None is majoring in the topics selected for the study.

The study simulated an exploration scenario, where the participant receives a list of recommendations while reading a Wikipedia article. There were three tasks per condition: two focused exploration tasks and a broad exploration task. For the former, participants had to find the five most relevant items for a set of two or three given keywords (Q1 and Q2). These tasks reflect the behavior of shifting information interests to a new topic while exploring. The broad exploration task (Q3), which consisted in finding five items relevant to a short text, reflects the need to clarify a textual description, building phrases to describe information needs. All combinations of *tool*, *#items* and *topic* were randomly assigned with balanced Latin Square. For each condition, participants filled a 7-point likert scale NASA TLX questionnaire for subjective workload assessment. In turn, the system recorded selected items and completion time per participant and task.

Workload, Completion Time and Performance.

Figure 8(a) shows that overall workload was significantly lower under the *U* condition, independently from *#items*, $F(1,23) = 35,254, p < .01, r = .2$. It can be observed in Figure 8(c) that this tendency applies to all dimensions. The results also revealed that even though users tended to consume all the time allotted (Figure 8(b)), the lower *subjective* temporal demand (Figure 8(c)) suggests that they felt significantly more relaxed when working with *uRank*.

To analyze performance, we aggregated the collections gathered by all participants and computed cosine similarity across *tool*, *#items*, *topic* and task (Q1, Q2 and Q3). Similarity values in Table 1 between collections produced with and without *uRank* (**U vs L**) denote that overall choices regarding relevant documents matched three out of four times ($M = .73$, $SD = .1$), across all task types and regardless of *#items*.

Collections produced with our tool for the two variations of *#items* (**U-30 vs U-60**) turned quite similar across all topics and task types ($M = .8$, $SD = .12$, with a minimum of .62). In turn, comparisons for the baseline UI with 30 and 60 items (**L-30 vs L-60**) denote greater overall diversity ($M = .67$, $SD = .16$, with a minimum of .33), particularly for broad search task (Q3) respect to focused search (Q1 and Q2). For full results refer to di Sciascio et al.[6].

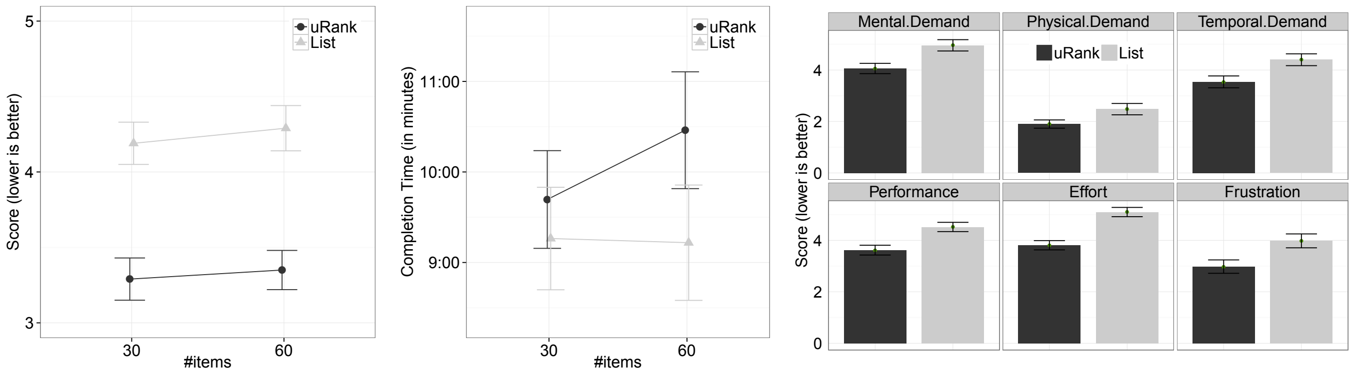


Figure 8. Study I Results. (a) Workload interaction lines show that *uRank* is significantly less demanding. (b) Time completion lines show a regularity towards using all available time. (c) Bar charts reveal lower workload for *uRank* across all dimensions. Error bars represent standard error.

Discussion and Limitations

It is a long-known problem that most users do not go beyond simple keyword search, despite the availability of more advanced UIs. Despite its highly controlled setup, the current study provided evidence that users actually engaged in exploration using *uRank*. Our tool is inherently more complex than a Google-like UI, yet users felt significantly less pressed by time and more confident about their performance. On top of that, participants were first-timers, i.e. they only had a few minutes of training prior to the evaluation tasks.

Nevertheless, the highly controlled and rather unnatural setting poses 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 towards common "search interests", which rarely represented 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.

USER STUDY II: ACTION ANALYSIS AND USABILITY

The second study explores the strategies users employ in exploratory search when working with *uRank* in a more natural setting, i.e. without rigid tasks, time-ups or uninteresting topics. Specifically, we were interested in addressing usability aspects and detecting usage patterns.

We sent invitations via e-mail to colleagues in the Computers Science field, which included a link to a Web form with all necessary guidelines. A total of 16 people accepted to take part in the study. A session started with a demonstrative video of the UI features. Thereafter participants had to open the *uRank* Web site in their browsers. They could chose a topic of their preference out of 7 collections, each with approximately 100 documents. The task consisted in freely exploring the collection and bookmarking interesting documents. We suggested between 5 and 10, although not as a strict condition.

The system recorded action logs: tag clicks, single and multiple drag-and-drop interactions, etc. After submitting the session data, users filled a survey consisting of: *(i)* questions addressing usability of specific UI components, and *(ii)* a standard usability questionnaire.

Table 2. User action summary

Type	Action	$M(SE)$	User Count
exploration	tag hover	135.06(21.03)	16
	tag click	2.63(0.50)	16
	keyword search	0.38(0.26)	2
control	ranking update	10.01(2.22)	16
	single tag dropped	4.44(0.68)	15
	multiple tags dropped	0.31(0.12)	5
	tag weight changed	4.13(1.53)	12
	tag deleted	1.13(0.43)	6
	reset	0(0)	0
drill-down	document click	74.81(26.57)	16
	document bookmark	4.63(0.81)	12
	document unbookmark	0.06(0.06)	1
	document watched	1.56(0.64)	9
	document unwatched	0.19(0.10)	3

Action Analysis

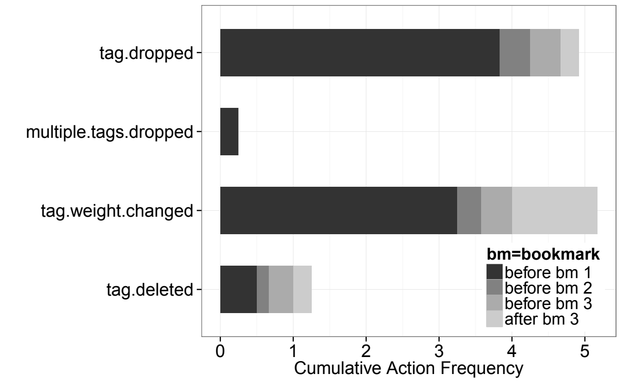
In this section we break down action log information according to action-type categories: exploration, control and drill-down. Table 2 presents a summary of recorded user actions.

We contrast action logs against user feedback provided for UI-specific questions. Responses were collected on a 7-point likert scale, where most questions were phrased in a positive tone (1=strongly disagree, 7=strongly agree), e.g. "Looking at the colors in keywords and bars, it was clear how the ranking was computed". The scale was inverted for negative-tone questions, e.g. "The position indicator in the document list was confusing" (1=strongly agree, 7=strongly disagree), so that all values can be interpreted as "higher is better".

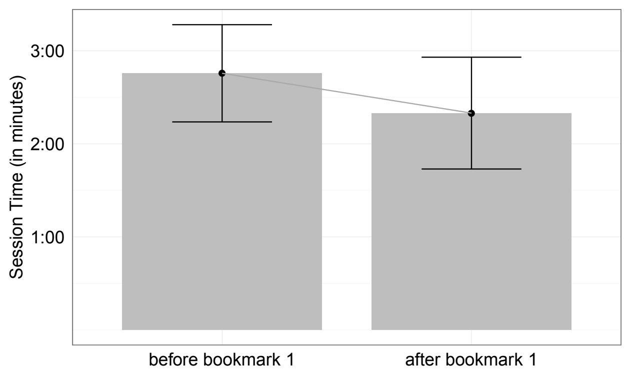
Exploratory Actions

Action logs show that users extensively hovered on keyword tags (135 times on overage) but in most cases did not click on tags to preview bearing documents and potential key phrases ($M = 2.63$, $SE = 0.5$).

Usage data for the keyword range slider reveal interesting results: only 4 participants interacted with it – with two of them tuning it 117 and 303 times, respectively –, but most



(a)



(b)

Figure 9. Study II Results. (a) Stacked bars for ranking updates distribution around 1st, 2nd and 3rd bookmark. (b) Bar chart shows average pre- and pos-1st-bookmark session time.

users reported that this feature was useful ($M = 5.09$, $SE = 0.35$). The keyword search feature was also scarcely used: only two participants actively searched for specific terms. This corresponds with previous studies: people preferred browsing the offered keywords to explicitly searching for them [12].

Control Actions

On average, users updated the document ranking 10.01 times. This evidences the role of interactive control features in engaging users in a trial-and-error search process. The rather low usage of tag clicks compared to the number of ranking updates ($2.64 vs. 10.1$) may suggest that users were more interested in immediately discovering the effect of manipulating a tag than in trying to predict it beforehand.

Most control actions corresponded to simple tag additions (4.44 times) and weight changes (4.13). These observations match user responses, as they admitted mostly choosing keywords one by one ($M = 5.82$, $SE = 0.38$) and frequently using the sliders to change keyword weights ($M = 4.77$, $SE = 0.37$).

Multiple drag and drop was only used by roughly a third of the participants (0.31 times), corroborated by the answers to the question "I mostly chose groups of keywords together" ($M = 2.45$, $SE = 0.39$). When asked about how easy it was to use this mechanism, responses were quite neutral ($M = 4.09$, $SD = 0.36$). Perhaps some participants found it too complicated (two of them actively reported so), but the majority just ignored it. 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.

Tag deletions remained in a low rate: two thirds of the participants never performed this action. The rest did it between 2 and 3 times (overall 1.13 times). None reset the ranking.

Drill-down Actions

8 out of 15 participants clicked on a specific document between 15 and 100 times (89 times on average). 4 outliers only inspected document abstracts between 1 and 5 times and 2 more than 300 times. In general, participants admitted that they often read document titles ($M = 6.09$, $SE = 0.2$), and sentences surrounding colored keywords ($M = 4.95$, $SE = 0.34$). The level of agreement to reading full abstracts was moderate ($M = 4.41$, $SE = 0.33$).

Surprisingly, 4 users did not bookmark any document. Excluding them ($N = 12$), users bookmarked 6.1 documents on average, which is not a discouraging number considering that the task did not impose a minimum amount. Participants did not revert their decisions: only one item was unbookmarked by accident and bookmarked again.

The questionnaire revealed that by looking at the colors of tags and bars it was clear for them how the ranking was computed ($M = 5.77$, $SE = 0.26$). Also, they chose their bookmarks mostly by looking at the titles and the ranking ($M = 5.27$, $SE = 0.27$), which denotes that users trusted the recommendations. Summarizing, the evidence suggests that the UI helped participants achieve a high level of self-perceived performance and that they felt confident about their decisions.

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: i) before the first bookmark, and i) between subsequent bookmarks. In the current analysis we only considered users that bookmarked at least one document ($N = 12$).

At first glance, Figure 9(a) reveals a strong tendency for tag selections to occur prior to the first bookmark event. All multiple tag drops were performed at this stage, while roughly 75% of all single tag drops fell therein ($M = 3.83$, $SE = 0.75$). As stated before, tag deletions were infrequent (overall, $M = 1.25$, $SE = 0.55$), but appear evenly distributed along all search phases. Weight changes also reach their peak before the first bookmark cut ($M = 3.25$, $SE = 1.24$), then tend to decrease towards the second ($M = 0.33$, $SE = 0.23$) and third bookmarks ($M = 0.42$, $SE = 0.33$), and finally a slight increment appears towards the end ($M = 1.17$, $SE = 0.79$). The distribution of actions depicted in Figure 9(a) suggests that after the first bookmark users were quite certain about the chosen keywords, although they fine-tuned the ranking minimally even after identifying three relevant documents.

This notion is supported by temporal distribution. Considering that an average session lasted approximately 5 minutes, Figure 9(b) reveals that pre- and pos-first-bookmark exploration were balanced (on average 2'45" and 2'20", respectively). Thus, the tipping point in a session was roughly marked by the occurrence of the first bookmark. Control actions in general were executed in the first half of a session. Afterward, 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.

Finally, 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. This can be validated by looking at the number of keywords per bookmark ($M = 4.32$, $SE = 0.61$), which did not differ significantly from the total number of unique keywords ($M = 5$, $SE = 0.65$).

Usability Analysis

In addition to questions for specific UI features, participants also filled a Software Usability Scale (SUS) questionnaire [4], a standard post-study questionnaire for subjective assessment of usability. Participants were not English native speaker, hence we chose a version with all positive-tone questions [32] 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. 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 84 ($SD = 9.4$). *uRank* falls in the 90-95 percentile range in the curved grading scale interpretation of SUS scores [32], 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 [21]. Adjusting multipliers for a 7-point likert scale (2.08 and 8.33, respectively), *uRank* scored an A for *Usable* ($M = 82.8$, $SD = 10.03$) and an A+ for *Learnable* ($M = 90.4$, $SD = 11$).

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 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.

Discussion and Limitations

This study revealed which parts of the tool proved most useful and easy to handle. Additionally, it provided some insights on how exploratory search unfolded: for example, control actions occurred mostly before the first bookmark, and then users dedicated 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 from 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 interests and further interactions. The short duration of an average session (5'05") corroborates this assumption.

We believe the only way to produce a realistic situation would be through a longitudinal study, e.g. 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 exactly how an optimal setup should be is out of the scope of this paper.

CONCLUSIONS AND FUTURE WORK

This paper introduced the reasoning line for the visual and interactive design of *uRank*, a visual analytics tool for exploratory search, along with incremental improvements and their rationale. Although originally planned as a recommending interface, it could be extended to generic search interfaces for not only unstructured data like text, but also for structured data. The latter would require that input data previously enriched with semantic information, tags or named entities.

We also presented two complementary user studies that support the design. The first study consisted in 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 important 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 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 a light on interaction paths and browsing strategies followed during exploratory search. Moreover, the tool received positive critics, as supported by the post-study questionnaires.

In the future we will pursue improvements in the UI, such as a more intuitive mechanism for multiple drag-and drop, and an overview component for navigating large lists. We also plan to extend text-mining methods by enriching keyword with synonyms and semantic relationships. Moreover, 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 1.

ACKNOWLEDGMENTS

This work is funded by the EC 7th Framework project EEXCESS (grant 600601). The Know-Center is funded within the Austrian COMET Program - Competence Centers for Excellent Technologies - of the Austrian Federal Ministry of Transport, Innovation and Technology, the Austrian Federal Ministry of Economy, Family and Youth and by the State of Styria. COMET is managed by the Austrian Research Promotion Agency (FFG).

REFERENCES

1. 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 (2002), 166–181. DOI: http://dx.doi.org/10.1057/palgrave.ivs.9500023

2. David M Blei and John D Lafferty. 2009. Topic Models. *Text Mining: Classification, Clustering, and Applications* (2009), 71–89. DOI: http://dx.doi.org/10.1145/1143844.1143859

3. Eric Brill. 1992. A Simple Rule-based Part of Speech Tagger. In *Proceedings of the Workshop on Speech and Natural Language (HLT '91)*. 112–116. DOI: http://dx.doi.org/10.3115/1075527.1075553

4. 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

5. Douglass R Cutting, David R Karger, Jan O Pedersen, and John W Tukey. 1992. Scatter / Gather : Browsing A Cluster-based Large Document Approach Collections to Scatter / Gather. *Sigir'92* (1992), 318–329. DOI: http://dx.doi.org/10.1145/133160.133214

6. Cecilia di Sciascio, Vedran Sabol, and Eduardo E Veas. 2015. uRank: Exploring Document Recommendations through an Interactive User-Driven Approach. In *Proceedings of the Joint Workshop on Interfaces and Human Decision Making for Recommender Systems, IntRS 2015, co-located with ACM Conference on Recommender Systems (RecSys 2015)*, Vol. 1438. 29–36. http://ceur-ws.org/Vol-1438/paper5.pdf

7. Debasis 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*. 1103–1104. DOI: http://dx.doi.org/10.1145/2484028.2484202

8. 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), 457–70. DOI: http://dx.doi.org/10.1109/TVCG.2013.242

9. 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), 2277–86. DOI: http://dx.doi.org/10.1109/TVCG.2013.173

10. 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 Trans. Intell. Syst. Technol.* 3, 2, Article 23 (Feb. 2012), 26 pages. DOI: http://dx.doi.org/10.1145/2089094.2089099

11. Mark Harrower and Cynthia A. Brewer. 2003. ColorBrewer.org: An Online Tool for Selecting Colour Schemes for Maps. *The Cartographic Journal* 40, 1 (June 2003), 27–37. DOI: http://dx.doi.org/10.1179/000870403235002042

12. M Hearst. 2011. User interfaces for search. *Modern Information Retrieval* (2011). http://people.ischool.berkeley.edu/~hearst/papers/mir_revision.pdf

13. 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, 59–66.

14. Jeffrey Heer and George Robertson. 2007. Animated transitions in statistical data graphics. *IEEE transactions on visualization and computer graphics* 13, 6 (2007), 1240–7. DOI: http://dx.doi.org/10.1109/TVCG.2007.70539

15. Orland Hoeber and Xue Dong Yang. 2006. The Visual Exploration of Web Search Results Using HotMap. In *10th International Conference on Information Visualisation, IV 2006*. 157–165. DOI: http://dx.doi.org/10.1109/IV.2006.108

16. Yuening Hu, Jordan Boyd-Graber, Brianna Satinoff, and Alison Smith. 2014. Interactive topic modeling. *Machine Learning* 95, 3 (2014), 423–469. DOI: http://dx.doi.org/10.1007/s10994-013-5413-0

17. Antti Kangasrääsiö, Dorota Glowacka, and Samuel Kaski. 2015. Improving Controllability and Predictability of Interactive Recommendation Interfaces for Exploratory Search. In *IUI*. 247–251. DOI: http://dx.doi.org/10.1145/2678025.2701371

18. Judy Kay. 2006. Scrutable adaptation: Because we can and must. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Vol. 4018 LNCS. 11–19. DOI: http://dx.doi.org/10.1007/11768012_2

19. Bart P. Knijnenburg, Svetlin Bostandjiev, John O'Donovan, and Alfred Kobsa. 2012. Inspectability and control in social recommenders. *Proceedings of the 6th ACM conference on Recommender systems - RecSys '12* (2012), 43. DOI: http://dx.doi.org/10.1145/2365952.2365966

20. M. Krishnan, S. Bohn, W. Cowley, V. Crow, and J. 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

21. 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

22. Christopher D. Manning. 2008. *Introduction to Information Retrieval*. Cambridge University Press.

23. Gary Marchionini. 2006. Exploratory search: from finding to understanding. *Commun. ACM* 49, 4 (2006), 41. DOI: http://dx.doi.org/10.1145/1121949.1121979

24. Tien N Nguyen and Jin Zhang. 2006. A novel visualization model for web search results. *IEEE transactions on visualization and computer graphics* 12, 5 (2006), 981–8. DOI: http://dx.doi.org/10.1109/TVCG.2006.111

25. John O'Donovan, Barry Smyth, Brynjar Gretarsson, Svetlin Bostandjiev, and Tobias Höllerer. 2008. PeerChooser: Visual Interactive Recommendation. *Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems* (2008), 1085–1088. DOI: http://dx.doi.org/10.1145/1357054.1357222

26. Denis Parra, Peter Brusilovsky, and Christoph Trattner. 2014. See what you want to see: Visual User-Driven Approach for Hybrid Recommendation. *Proceedings of the 19th international conference on Intelligent User Interfaces - IUI '14* (2014), 235–240. DOI: http://dx.doi.org/10.1145/2557500.2557542

27. M.F. Porter. 1980. An algorithm for suffix stripping. *Program: electronic library and information systems* 40, 3 (1980), 211–218. DOI: http://dx.doi.org/10.1108/00330330610681286

28. Ramana Rao and Stuart K Card. 1994. The table lens. In *Proceedings of the SIGCHI conference on Human factors in computing systems celebrating interdependence - CHI '94*. ACM Press, New York, New York, USA, 318–322.

29. Manuela Rauch, Werner Klieber, Ralph Wozelka, Santokh Singh, and Vedran Sabol. 2015. Knowminer Search - a Multi-Visualisation Collaborative Approach to Search Result Analysis. In *Information Visualisation (iV), 2015 19th International Conference on*. 379 – 385. DOI: http://dx.doi.org/10.1109/iV.2015.72

30. Harald Reiterer, Gabriela Tullius, and TM Mann. 2005. Insyder: a content-based visual-information-seeking system for the web. *International Journal on Digital Libraries* (2005), 25–41. http://link.springer.com/article/10.1007/s00799-004-0111-y

31. Francesco Ricci, Lior Rokach, and Bracha Shapira. 2011. Introduction to Recommender Systems Handbook. In *Recommender Systems Handbook*, Francesco Ricci, Lior Rokach, Bracha Shapira, and Paul B. Kantor (Eds.). Springer, 1–35.

32. Jeff Sauro and James R. Lewis. 2012. *Quantifying the User Experience*. Morgan Kaufmann.

33. Christin Seifert, Johannes Jurgovsky, and Michael Granitzer. 2014. FacetScape : A Visualization for Exploring the Search Space. In *Proceedings 18th International Conference on Information Visualzation*. 94–101. DOI: http://dx.doi.org/10.1109/IV.2014.49

34. Christin Seifert, Barbara Kump, Wolfgang Kienreich, Gisela Granitzer, and Michael Granitzer. 2008. On the Beauty and Usability of Tag Clouds. *2008 12th International Conference Information Visualisation* (July 2008), 17–25. DOI: http://dx.doi.org/10.1109/IV.2008.89

35. Guy Shani and N Tractinsky. 2013. Displaying relevance scores for search results. *Proceedings of the 36th international ACM SIGIRâĂŹ13* (2013), 901–904. http://dl.acm.org/citation.cfm?id=2484112

36. Anselm Spoerri. 2004. Coordinated Views and Tight Coupling to Support Meta Searching. In *Proceedings of Second International Conference on Coordinated and Multiple Views in Exploratory Visualization*. 39–48. DOI: http://dx.doi.org/10.1109/CMV.2004.5

37. Marc Streit and Nils Gehlenborg. 2014. Bar charts and box plots. *Nature methods* 11, 2 (Feb. 2014), 117. http://www.ncbi.nlm.nih.gov/pubmed/24645191

38. K. Swearingen and R. Sinha. 2001. Beyond Algorithms Beyond Algorithms: An HCI Perspective on Recommender Systems. *ACM SIGIR 2001 Workshop on Recommender Systems (2001)* (2001), 1–11. DOI: http://dx.doi.org/10.1.1.23.9764

39. 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

40. 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* (2013), 351. DOI: http://dx.doi.org/10.1145/2449396.2449442

41. Michelle Q. Wang Baldonado, Allison Woodruff, and Allan Kuchinsky. 2000. Guidelines for using multiple views in information visualization. *Proceedings of the working conference on Advanced visual interfaces (AVI)* (2000), 110–119.

42. Matthew O Ward, Georges Grinstein, and Daniel A Keim. 2010. *Interactive Data Visualization: Foundations, Techniques, and Application*. A. K. Peters, Ltd.

43. C Ware. 2013. *Information visualization: perception for design* (3rd ed.). Elsevier.

44. Ka-Ping Yee, Kirsten Swearingen, Kevin Li, and Marti Hearst. 2003. Faceted metadata for image search and browsing. *Proceedings of the conference on Human factors in computing systems - CHI '03* (2003), 401–408. DOI: http://dx.doi.org/10.1145/642679.642681