BROWN UNIVERSITY

Exploratory Search in WorkTop

Seth Goldenberg

sethg@cs.brown.edu

February 9, 2012

Introduction

When browsing or searching through large document sets, faceted search and browsing interfaces are effective tools for finding specific documents or facets. Often, users must navigate a collection of documents with little or no understanding of the subject matter or theme of the collection. They may not know which search terms, documents, or facets are relevant to their interests. This particular scenario is commonly referred to as *exploratory search* in the human-computer interaction (HCI) community.

Beginning to study or research a new field can at times be daunting. Many people start simply by entering the most basic of search terms, like “human-computer interaction” or “exploratory search”, into a search engine such as Google or Bing and see where the results take them. While this technique may be sufficient in many cases, it forces users to be overly reliant on information retrieval algorithms (albeit very effective ones) to locate important sources. Even then, too much interaction is required to find salient information. What happens if the user misses a prominent expert’s opinion on the second or third page of the results? We would argue that said user does not get a full picture of the field.

User interface researchers have devoted much of the past decade developing faceted classification systems with a focus on exploratory search. These systems allow for the input of targeted search terms for those well versed in a repository’s subject matter, while users with less focused goals can find information by browsing facets. This balance of capabilities is intended to accommodate all users, but it fails to truly assist new users in exploring an unfamiliar area. In many systems, users may view lists of facets but are often required to perform many operations to find connections between them. Users must construct their own mental maps through tedious interactions. If the repository changes over time, as the web does on a daily basis, any understanding of a subject or event can be lost.

In this work, we will present two interfaces that aid in exploratory search by focusing on exposing relationships in document metadata. The first interface visually arranges documents by metadata analysis and is intended to assist in exploring smaller document sets. The second interface focuses on visualizing connections in larger collections using node-link diagrams. Each interface balances users’ divergent needs to both explore tangentially related facets and to narrow their search focus.

WorkTop

The search interfaces described in this work were implemented within a system called WorkTop. WorkTop is a desktop environment designed to help users perform fundamental scholarly tasks and encourage collaborative learning and knowledge creation [1]. Users import documents of various types (e.g., images, PDFs, movies, rich text documents) into a centralized shared repository dedicated to a specific subject or topic matter. They can then annotate documents with metadata, keywords, and bi-directional fine-grained hyperlinks within and between documents, all of which are visible to the other users of the repository. WorkTop also allows users to perform faceted searches on the document metadata. In addition to the visualizations described in this paper, there are several other visual arrangers that are used to display search results. We will focus on those related to the area of exploratory search.

Users entering WorkTop for the first time may have little familiarity with the subject matter being explored and want an overview of the repository's contents before jumping in themselves. Those more familiar with the environment may want a summarized view of the work being done. These interfaces help users achieve both of these goals. While the visualizations were designed with WorkTop in mind, they are intended to help achieve sensemaking of any collection of documents. However, some of the aspects of this research are tailored to collaborative environments where users have some control over the contents of the repository and interact with the system on an ongoing basis. We make note of features that leverage these traits of WorkTop when appropriate.

Terminology

In this paper, we employ terminology used by the HCI community. Before jumping in, some of the important terms should be explained. A *category* describes a collection of documents that share a descriptor or attribute in common, typically a metadata field value (e.g., a specific author name, keyword, publication date) or some computed value (e.g., importance of document). A *facet* is a collection of typically closely related categories. For example, a location facet for a database of restaurants would partition the set of restaurants into different categories of cities, where each category corresponds to a specific location attribute value, i.e., a specific city such as Providence. In WorkTop, facets correspond to metadata fields such that the facet categories correspond to specific attribute values within those fields.

We evaluated and tested our interfaces using a repository of CHI publication metadata. Throughout this paper, we will be providing examples related to HCI publications and using facets that align with the metadata fields in that repository, including "Keywords", "Authors", and "Publication Date."

Related Work

Researchers have devoted a great deal of effort in the past two decades towards the area of faceted search and browsing. The Flamenco system [2] was an early demonstration that combined faceted browsing with comprehensive query capabilities and showed a major improvement over pure keyword search or pure categorization. Microsoft's PivotViewer is a public library for Silverlight that lets developers perform faceted searches on their own data [3]. True to its name, the PivotViewer interface facilitates easy pivoting on facets and their categories. FacetLens, and its predecessors FacetMap and PaperLens provide users with a faceted classification system [4] [5] [6]. In addition to a browsing and search controls, FacetLens employs bar graphs to let users identify and compare trends over time. FacetLens also enables pivoting on facet categories, allowing the navigation of faceted datasets using the relationships between documents.

Revealing relationships within a dataset is a simple yet effective summarization technique employed by several other systems. One such system is EdgeMaps, which uses node-link diagrams (NLDs) to show connections between entities in a dataset in a manner similar to our own system [7]. Unlike our own work, EdgeMaps also depicts implicit relationships that are derived from multidimensional data based on shared properties and similarity measures.

Contributions

Of all the faceted classification systems encountered in our research, we consider the FacetLens system one of the best, especially for exploratory search. Our work is largely a departure from FacetLens because we felt we could not significantly improve upon their existing system. Like FacetLens, we seek to expose connections between facets and their values, but we do so without an interface for browsing facets. Instead, we focus on making these connections as obvious as possible by minimizing the amount of interaction to find them. Nonetheless, there are concepts and techniques in our work that were inspired by FacetLens. Browsing in our system is greatly improved by the ability to pivot on a facet or category. We also let users enter targeted searches to refine their searches. When a category is hovered over in FacetLens, attribute values that share at least one document in common with the selected value are highlighted with a brushing effect. We use the same mechanism for a similar purpose in our interfaces (Figure 3). The idea to test and evaluate our system on the CHI repository came from their publication as well.

Node-link diagrams have long been used to in information visualization. EdgeMaps is a recent system that bears the most resemblance to our own work. EdgeMaps utilizes NLDs to diagram relationships between nodes, but the similarities end there. They present three demos[1] that highlight the influences that painters, musicians, and philosophers had on one another. Vertex sizes are determined by an individual's significance in his field, which we borrow in our own work. Unlike our system, edges between vertices are directed, where direction indicates who influenced whom. The authors introduce two graph layouts for the placement of vertices. The "Similarity Map" layout uses a multidimensional scaling algorithm to position similar nodes near one another in two-dimensional space. In the artist demo, Picasso and Matisse are adjacent to one another, but both are far from Monet. The "Timeline" layout displays all nodes on a single line in chronological order of the person's lifetime, which lets users see influence relationships much clearer. The authors also built upon their earlier research on graph presentation to create a diagram that is both visually appealing and easy to parse. It is worth noting that the data in WorkTop can be much more diverse; documents generally do not correspond to such distinct facets like painters or musicians, making their layouts difficult to implement in our system. Our interfaces allow for a great deal of interaction and layout customization to foster exploration, whereas EdgeMaps is primarily a fixed browsing interface. We believe this distinguishes our work from existing systems like EdgeMaps.

[1] http://mariandoerk.de/edgemaps/demo/

Figure 1 – Standard grid arrangement of a folder containing image and PDF documents in WorkTop

Smart Grouping

Search engines like Google and Bing traditionally display results in list format, one document after another ranked by a relevance metric. Other interfaces, like a filesystem browser or the standard view within WorkTop, arrange documents in a grid format, using thumbnails to represent documents (Figure 1). When dealing with large or unfamiliar document sets, sifting through the results can be a tall order. Layouts that only present documents according to a single metric like relevance or date often fail to assist users in navigating search results. In this section, we introduce *Smart Grouping,* a new visualization to aid in the task of locating documents of interest.

Algorithm

Smart Grouping performs an analysis of the metadata of documents in a collection and determines which breakdowns are potentially interesting to the user. Each document in the system can be assigned metadata values. For example, in a repository of CHI papers, you can expect to see documents tagged with keywords such as "information visualization" or "touch gestures." Documents can have multiple values for the same field, like multiple authors for a single paper. *Smart Grouping* brings these values forward in the visual arrangement of the results by automatically grouping documents with common values.

To determine which metadata fields and values to use for grouping, we use a simple heuristic to score each facet in the document set (e.g., "Keywords", "Date Published") and rank them by their resulting score. At the time of this writing, WorkTop does not have typed metadata; each metadata value is stored as a string. Our system does attempt to infer more complex types like dates in some cases. However, users may enter any string for any metadata field in the repository. As a result, we did not consider using clustering algorithms or machine learning techniques that would require data to be of continuous types (i.e. numbers, dates). Instead, metadata values are treated as strings and are only grouped when equal. When examining a date field, “9/7/2009” would be treated as a different value than “09/07/09”. We see this as a limitation in WorkTop and hope to change this in future iterations.

In our heuristic, each facet is assigned a score that is calculated using the sizes of categories within that facet. The scoring system awards points to facet breakdowns that meet the following criteria:

- **Diverse facets -** Facets that contain more than two distinct categories are given a point. As an example, a point would be awarded to the “Authors” facet if there were at least three authors in a set of search results.
- **Balanced facets** - Any facet with a single category that contains more than three quarters of the document set is penalized a point. For example, the set of CHI documents are all PDFs, which makes that category uninteresting in exploration.
- **Larger categories** - In contrast with the previous criterion, we also wish to avoid facets in which many categories have one or a few items. Facets receive a point if at least half of the categories contain more than one document. Many of keywords from the CHI repository are incredibly specific (e.g., “high degree freedom-of-input”, “3D Fitts’ Law”). In a set of search results with many unique keywords, the heuristic would favor showing authors who have written multiple papers.
- **Avoiding sparse facets** - It is possible for a document not to have a value for a metadata field. We award a point to facets where the size of largest category is greater than the number of documents that do not belong to any category. It is difficult to come up with an example for the CHI repository since each metadata field is present on all documents. In a WorkTop repository where users import multiple types of documents, certain metadata fields may only apply to specific document types. For example, “Director” metadata would only be present on a movie. In a collection of mixed documents, this facet would not be awarded a point.

The facet with the highest score is presented to the user. The same heuristic is applied to documents of each category of this facet, forming nested categories. We only create one level of nested categories to avoid cluttering the interface and overwhelming the user with too much information. To highlight the more prominent categories, categories below a certain size threshold are merged into a “miscellaneous” category. The next section will explain how the facets and categories are displayed.

Display & Interaction

The goal of Smart Grouping is to provide easy access to documents while exposing relationships in the search results. Documents are represented by thumbnail summarizations of their contents, which limits the number of documents that can be displayed at a time. As a result, the interface described is intended for search results of less than 100 documents. Categories and the nested categories they contain are labeled and serve as containers for the thumbnails. A thumbnail can be clicked to open the full document. If the facet breakdown selected by our algorithm is not sufficient, users can select a different facet from a menu. We show the facet options ordered by their score so users can see which facets are deemed more interesting by our heuristic.

Figure 2 shows the Smart Grouping view applied to all CHI publications authors by Tovi Grossman. The heuristic has determined that "Publication Date" is the most interesting facet to display for this set of documents. We can see that Grossman has managed to publish a large number of papers in the past few years. The categories within each "Publication Date" category show authors who have collaborated with Grossman. George Fitzmaurice and Ravin Balakrishnan appear in multiple years, showing that they have likely worked with Grossman on several projects. Currently, categories are ordered by size. With better support for metadata date values in WorkTop, we could arrange categories by date and illustrate trends over time more easily.

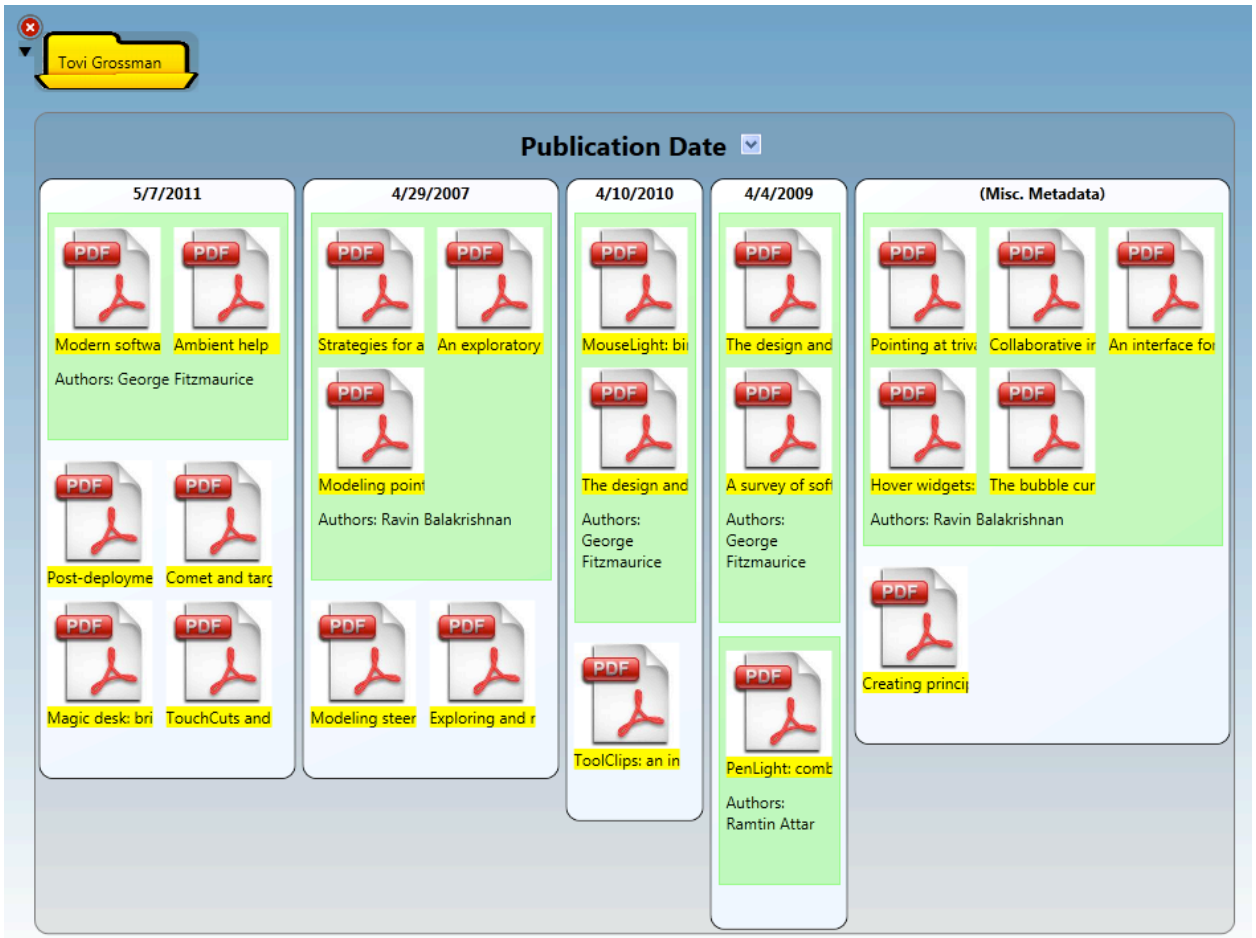

Figure 2. The Smart Grouping view is applied to the results of a faceted search on the "Author" facet for the "Tovi Grossman" category. Clusters are divided on the "Publication Date" facet. Categories within the categories show authors who co-authored papers with Grossman. The containers for nested categories are color coded by facet. For example, all categories in the "Authors" facet are green. It should be noted that the first page of a PDF is generally used as the thumbnail for documents. Unfortunately, we only have access to the CHI document metadata, not the documents themselves. We use a generic PDF icon to represent the documents in this view.

In other faceted browsing systems like FacetLens and PivotViewer, users can see the top authors, keywords, and even yearly breakdowns for Tovi Grossman. It is not clear, however, that Ravin

Balakrishnan co-authored 3 of Grossman's 5 papers from the 2007 conference. These systems display information on a single facet at a time, but they do not highlight connections between multiple facets. Users must discover interesting facets on their own. The Smart Grouping view aims to point out interesting breakdowns with minimal interaction.

We have shown how Smart Grouping can help users make sense of the documents returned from a search query. But what if users want to use the knowledge gained from performing a search to find related documents? We introduced a widget within the view to enable this sort of tangential exploration of the repository (Figure 3). Each category container has an icon control, which when clicked, opens a statistical overlay that shows the list of categories that have at least one document in common with the selected category. The number of matching documents in the folder is displayed next to category name. Moving the mouse over a row in the list highlights documents in the current set of results that match the category and displays thumbnail previews of documents outside of the results. Dragging out a row opens a new folder of documents matching the selected category. This allows users to view related documents without disturbing the Smart Grouping view.

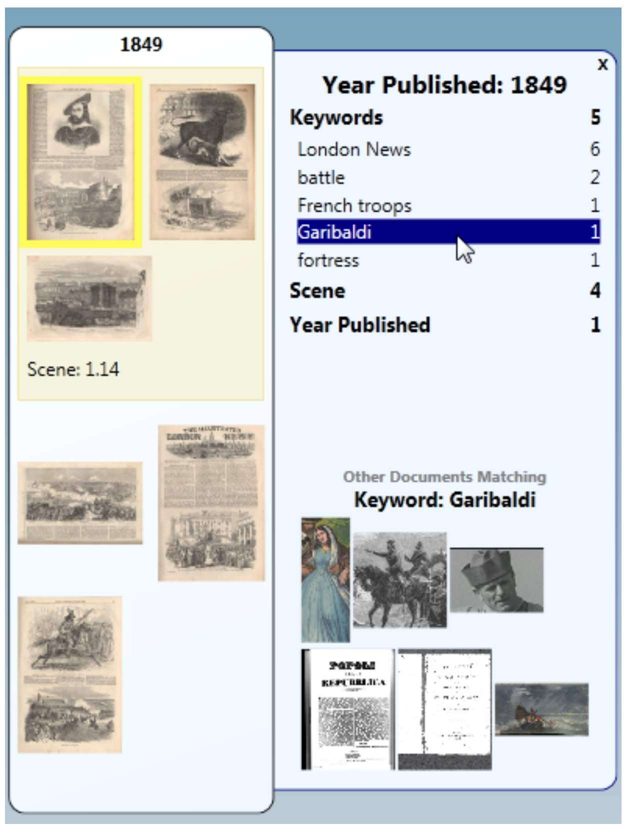

Figure 3. Statistical overlay in Smart Grouping view. The count of documents in the search results that match each category term is listed next to the text for that category. Moving the mouse over a category term highlights thumbnails of any document that is labeled with the selected term. The icon control to open the statistical overlay is only visible when the cursor is over the category container.

Whereas many faceted classification systems are designed for browsing and searching immutable datasets, WorkTop encourages the addition of metadata to all documents. To better facilitate these capabilities in WorkTop, thumbnails can be dragged to the category controls to apply the corresponding metadata value to a document. The previous controls to add metadata in WorkTop required users to open the full document and then open a widget to select metadata values to apply. The new drag and drop interface in Smart Grouping speeds up the process of annotating documents considerably.

Even though Smart Grouping adds valuable information to the layout of document thumbnails, the visual arrangement may still prove to be too overwhelming for users to parse. Users who wish to reduce clutter in the view can drag documents on top of one another to create stacks. Creating a stack condenses the thumbnail icons into the space of a single thumbnail, freeing up screen real estate for other documents or folders (see Fig. 4 where the stack appears third from the top left in the 1860 category. A tooltip shows the thumbnails of individual documents contained in a stack, and clicking on the stack opens all contained documents to their full view. Stacks are meant not only as a tool for improving layout but also as a means of grouping similar documents that should be viewed together. Stacks created by a user are seen by all users of the repository, essentially serving as a collaborative annotation. Users who do not wish to see a set of documents stacked can right-click and select an option to see the thumbnails displayed separately in the future.

Figure 4. A stack of 3 documents and a tooltip revealing the stack thumbnails. The "1 more" label below the stack indicates that there is one additional document in the category "Scene: 2.34" that is not being displayed.

When there are too many documents to fit within the full window size, the facet and subgroup controls display a subset of thumbnails. Since there is no ordering or ranking to documents in these controls, there is no distinguishing feature that can be used to select which thumbnails to be displayed. The thumbnails displayed are selected at random in these instances, and the number of documents being hidden is listed below the thumbnails ("*n* more"). Although this is sufficient to render this view usable, Smart Grouping was meant to highlight individual documents within a smaller collection. In the next section, we introduce another view that is better suited for summarizing larger document sets.

Relationship Graph

Summarizing a large set of documents is a challenging task for designers and software developers. How do you give users a bird's eye view of hundreds or thousands documents? Google and Bing both assume that several excellent query matches are better than having to parse hundreds of good matches. They present paginated links ranked by their relevance to the search query. Facet classification systems like FacetLens show the largest categories in each facet. These techniques help users find specific documents that may be relevant to them but fall short in giving users an overview of the repository. In this section, we introduce the *Relationship Graph* view, a new visualization designed specifically for helping users understand the contents of a document collection.

The Relationship Graph view uses node-link diagrams (NLDs) to display important categories and the connections between them. The core aspect of the view is an undirected graph, where categories are represented as vertices and the connections between them as edges. Two categories are connected if at least one document matches both categories. For example, if a document is tagged as having the keyword "gestures" and author "Robert Zeleznik", those categories are connected and an edge is drawn between the vertices.

To generate the graph to display, we use the concept of a focus field. First, the occurrences of each metadata value are tallied. We select the facet with the most categories in the top results to use as the focus field. We then select the top ten categories of the focus field to be displayed in the graph. The categories that share the most connections with the focus field categories comprise the other vertices of the graph. Vertices in the graph are color-coded by facet; a legend indicating the color mapping is displayed below the graph. The metadata value text is displayed in the vertex control. Focus field vertices are emphasized with bold text. In a manner reminiscent of tag clouds, vertex font sizes correspond to the number of documents matching a category. This draws more attention to more popular categories. The number of documents shared by connected categories determines an edge's thickness. Thicker edges indicate a stronger connection. When the number of overlapping documents between categories drops below a threshold (e.g., one), a dotted line is used instead of a solid one to indicate a weaker connection.

We refer to figure 5 to illustrate an example of a Relationship Graph view. In this example, "Keywords" is the focus field, and "Authors" is the only other facet present in this dataset. The graph in this view highlights the most popular topics from the SIGCHI 2011 conference. By examining the vertex sizes and edge weights, a user can tell that "Privacy" was a popular topic of research that year. Lorrie Faith Cranor and Lujo Bauer published several papers on the subject, and some of them, if not all of them, were written together. In FacetLens, users can see a list of the top keywords of all documents but must mouse over or click each keyword to see authors who have contributed papers on that keyword. In contrast, the Relationship Graph view exposes these trends with minimal user interaction.

The focus field can be changed by accessing a menu in the legend below the graph. In Figure 6, the focus field has been switched to “Authors”, showing the most published authors and the topics to which they contributed. Changing the focus field causes new trends to appear. Jacob O. Wobbrock was one of the most published authors in 2011. Since the topics he wrote about were not as popular, he did not appear in the initial view when the “Keywords” facet was the focus field. From this view, we can see that Wobbrock worked on subjects like multitasking that were somewhat popular, yet he also worked in areas like accessibility that were not as commonly researched.

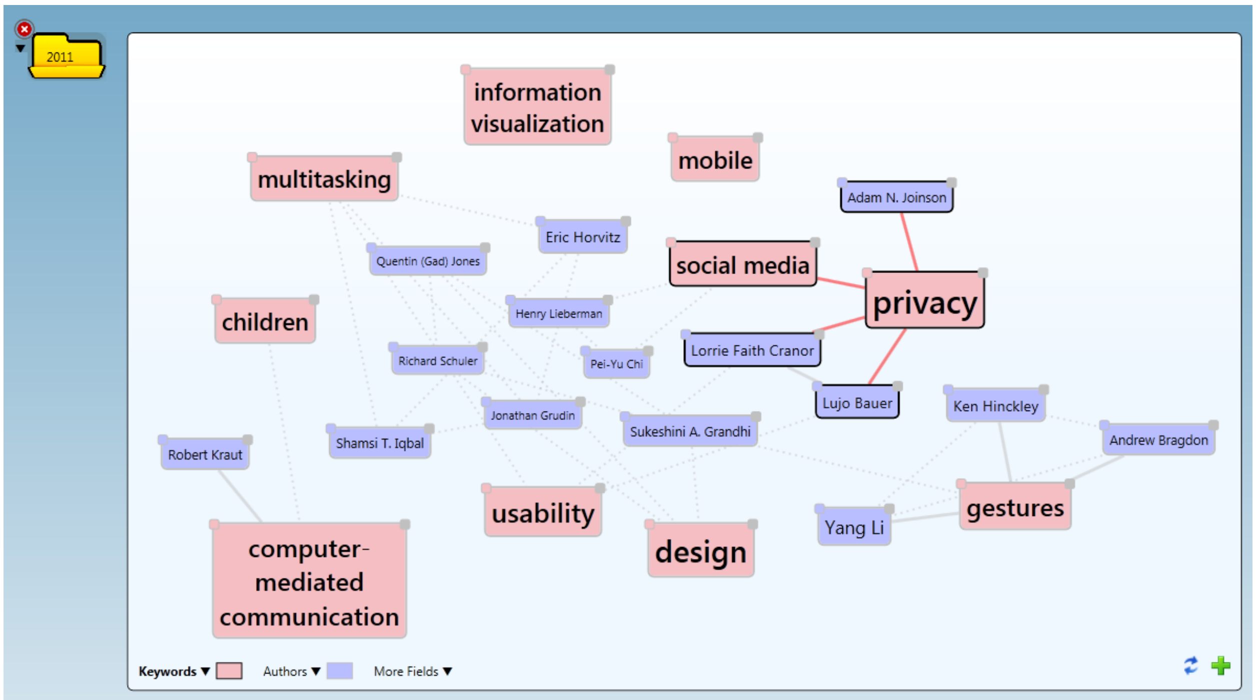

Figure 5. Relationship Graph view of CHI 2011 publications with “Keywords” as the focus field.

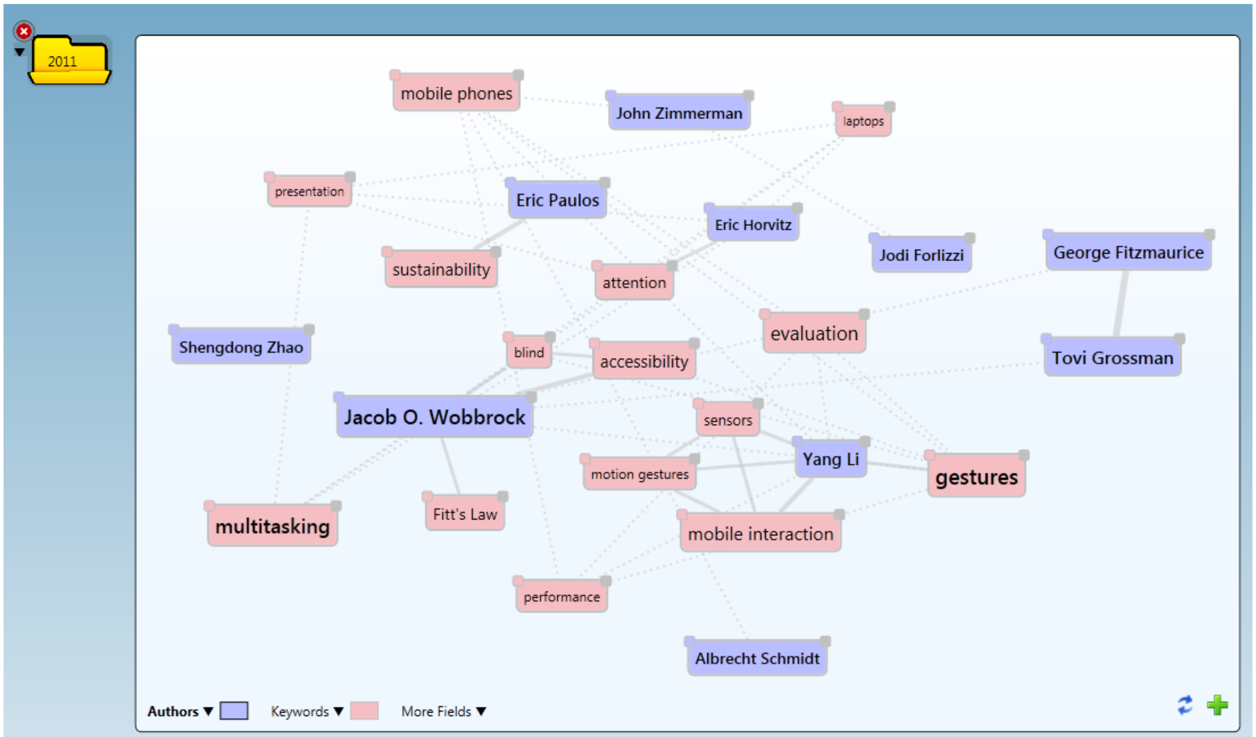

Figure 6. Relationship Graph view of CHI 2011 publications with “Authors” as the focus field.

In our research, we aim to minimize the work performed by users to uncover information relevant to them. Despite our efforts, tools are needed to explore beyond the initial graph presented. We provide several options for users to expand the existing graph to find other categories and connections. Each vertex control has a gray button at the top-right corner that is only visible when the vertex has connected categories that are not being displayed in the graph. Clicking this button opens an overlay showing a list of categories that can be connected to the vertex (Figure 7). Users can drag them out to add them to the graph. For convenience, users can click and drag the button directly to add the category with the next most connections to the graph. Holding the shift key and dragging will add all of the connected categories to the graph. The view also provides a type-ahead search control to let users circumvent exploration in favor of a more targeted search (Figure 8).

Figure 7. A Relationship Graph view vertex with the overlay of connected category terms that can be added. The square on the top-left corner of the vertex can be clicked to open a menu to change vertex's color.

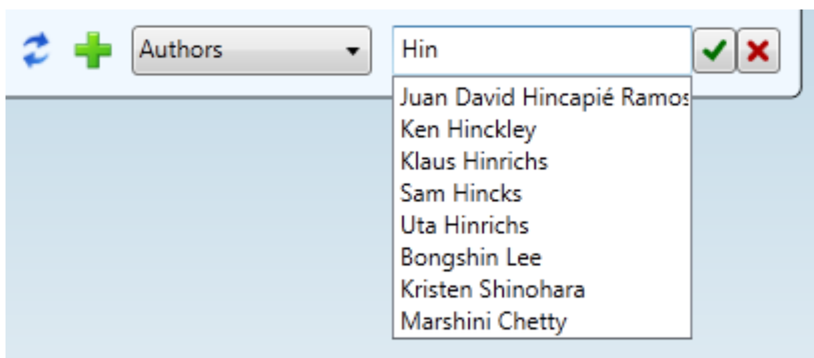

Figure 8. Type-ahead search control. The facet being queried can be changed by selecting from the dropdown. The suggestions are tailored to the selected facet.

Users need to be able to focus their explorations by removing uninteresting facets. Right clicking on a vertex reveals options to remove the vertex or to remove all of the vertex's connected components. Even though we have made adding and removing vertices to the graph as simple as possible, there is still a need for more ephemeral searching without losing an earlier state of a graph. Users can pivot on a specific category with a double click. Pivoting will add to the graph all categories connected to the vertex and remove all those that are not directly connected. Holding shift while double clicking performs the same operation but preserves all vertices with a path to the pivot vertex, keeping a larger portion of the graph intact. Upon pivoting, a navigation control appears above the graph, letting users move forward and back through pivot states or to return the original graph layout prior to pivoting (Figure 9). The control indicates the facet and category of the pivot vertex. The pivot vertex is colored yellow to draw the user's attention. To keep track of other vertices of interest, users can open a color picker widget by clicking the square at the top left corner of the vertex. The color selected for the category overrides the facet's color and is preserved when pivoting or manipulating the graph.

As the Relationship Graph view focuses on revealing connections in a document set rather than showing individual documents, the pivot control is the natural place to let users access documents matching a category. The folder icon can be dragged to the area around the graph to open a folder of the matching documents. The folder's contents can be viewed using the Smart Group view or any of the other folder arrangers in WorkTop.

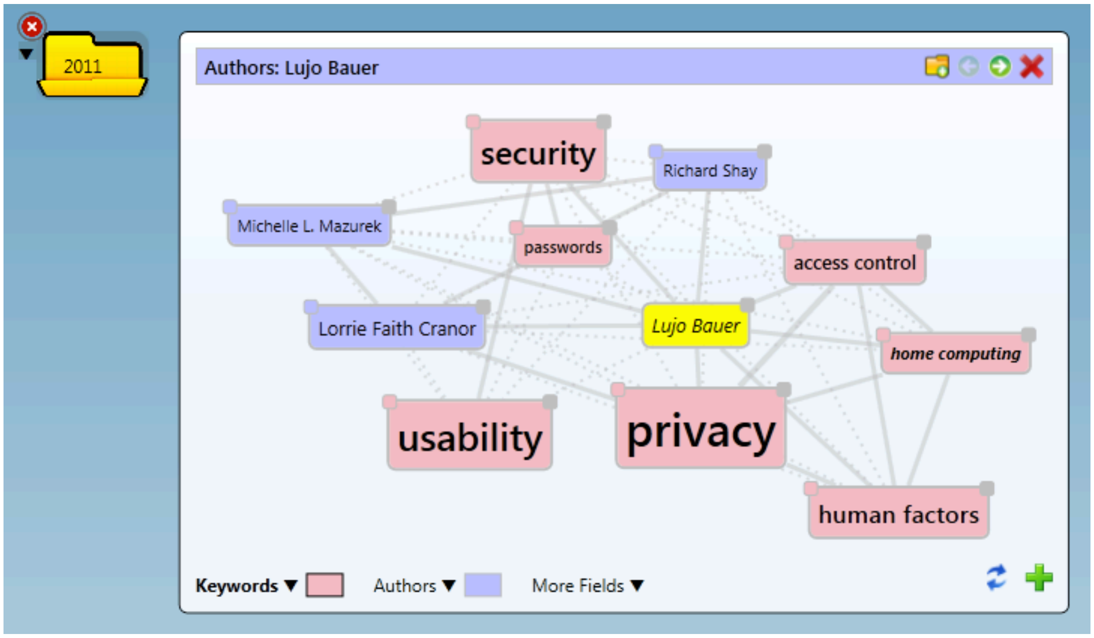

Figure 9. Relationship Graph view pivoted on author Lujo Bauer. Pivot controls can be found above the graph. The forward and back icons enable users to page through pivot states. Clicking the 'X' icon discards all pivot states and reverts the graph to its layout prior to pivoting.

Time is an important aspect of any set of documents. We noted earlier that WorkTop does not have any special handling for dates. We work around this shortcoming of the current implementation and find date fields in the metadata that can be parsed. If most of a facet's corresponding metadata values can be parsed as dates, we assume the facet is a date field. Clicking an icon in the Relationship Graph view reveals a date range slider that can be adjusted to show which categories contain documents that fall within the selected range (Figure 10). The opacity is lowered on category terms whose documents do not fall in the range. FacetLens does an excellent job displaying trends over time in document sets. In their system, users can view bar graphs for any category in repository. Bar graphs can be stacked for comparison of multiple categories. Admittedly, this is an area where FacetLens provides a better solution than the Relationship Graph view. We have only begun to explore this area and see great promise for further research.

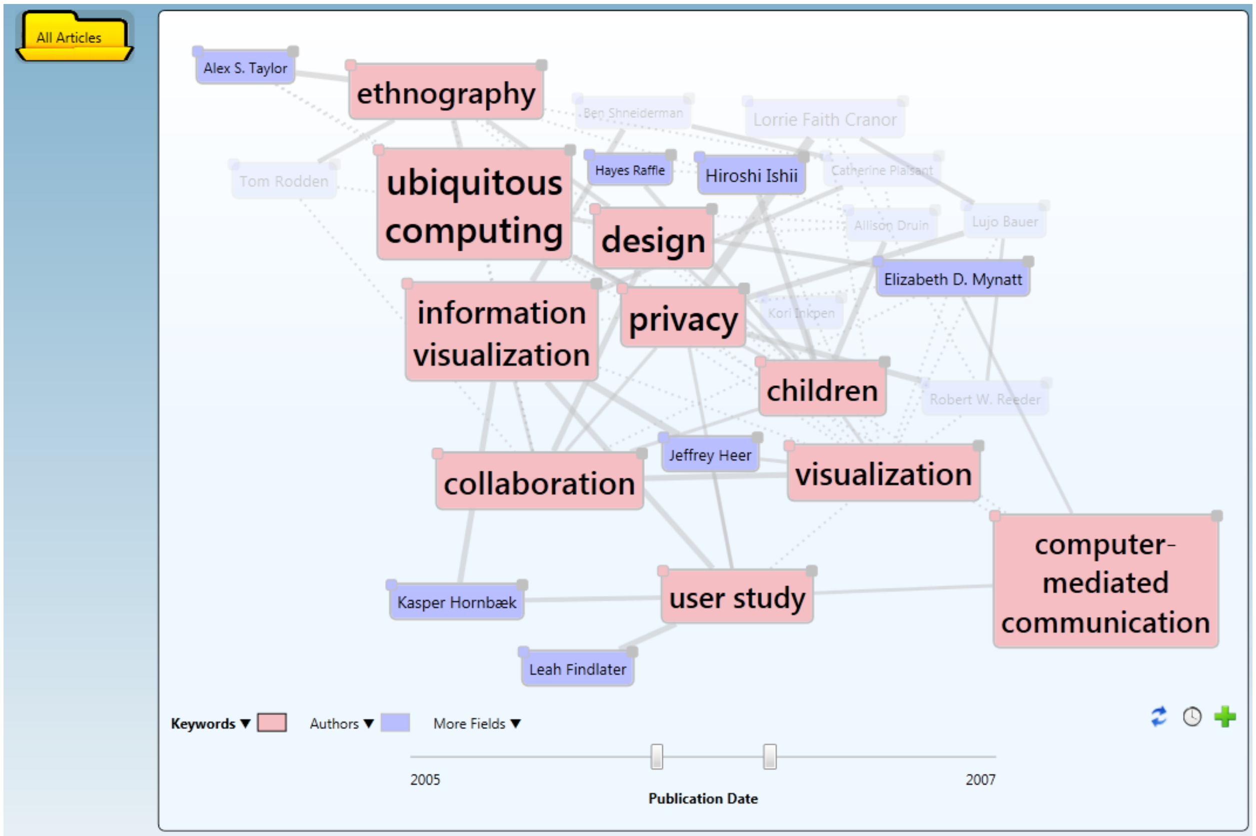

Figure 10. The date range slider shows which categories have documents that were published between 2005 and 2007. Authors and keywords that are not present in the timespan are given a lower opacity.

Most exploratory search interfaces do not support the saving of search results; users perform their searches, find what they need, and then get out. WorkTop, on the other hand, strives to foster collaborative research over a longer timeframe. To support this goal, all searches are saved and displayed in the UI as folders. The contents of the folders are automatically updated as documents and metadata are added to the repository. When users take the time to use the Relationship Graph view to explore one of these folders, the layouts are preserved so they can return to the same graph when the folder is opened again. This view not only serves as a means of exploring trends and facets within the repository but also as a mental map on a particular topic. The Relationship Graph can also display a snapshot of the entire repository, giving a glimpse into what types of documents and metadata tags each user has added. A summarization like this could prove invaluable to a professor surveying the work of a class or a researcher leading the work of her research team.

Limitations & Future Work

We feel that the work described in the previous section introduce a novel approach to exploratory search. We noted some of the features that are ripe for further development, but there are also a handful of other interesting research areas to be explored.

A unique feature of WorkTop is its “infinite canvas” layout. Unlike traditional WIMP interfaces, folders and documents in WorkTop are not locked to set window size. Users can pan across the desktop to access files hidden off screen. As a result, visualizations in WorkTop are not restricted to fixed dimensions. While the Relationship Graph view never grows to an unreasonable size, the Smart Grouping view often does not fit within the confines of a large monitor when displaying many documents. We attempt to hide thumbnails when the number of documents to display grows too large, even though this view is not intended to handle such large collections in the first place. Category controls, which contain the thumbnails, are stacked horizontally but vary in height, resulting in gaps around the edges of the view. The experience of using Smart Grouping would be improved if more consideration were given to the sizing and positioning of facets. Tree maps are excellent for visualizing hierarchical data in a fixed space and might prove to be good starting point for this research.

Smart Grouping serves as an improvement over the standard list or grid views of documents, typically seen in search engine result pages and file system browsing interfaces, yet it is not a huge departure from existing interfaces. We see more promise in the Relationship Graph view, which according to our assessment of the current research, is the more novel of the two visualizations. The Relationship Graph view also leverages more of WorkTop’s unique capabilities than Smart Grouping, like the ability to share folder layouts with other users.

We used the open-source graph layout library Graph# to create the graph control. We customized the styling and some of the behaviors to make the control fit our needs. The library also provides layout algorithms for positioning vertices. From the handful of algorithms, we found a force-directed placement (FDP) algorithm that created a clean layout and spaced vertices without wasting screen real estate. We tweaked the algorithm to minimize layout disturbances when vertices and edges were added to the graph. The modified FDP algorithm succeeds in creating an excellent initial graph that users can manually adjust to their liking. The ability to customize the graph and move vertices can be improved with an algorithm that takes into account information about the facets themselves. For example, trends may be more conspicuous if vertices of the same field were to snap to the same horizontal or vertical axis. As discussed earlier, EdgeMaps illustrates an example of using metadata to create a semantic spatial layout.

The graph can become cluttered when too many facets are added. We make users responsible for keeping the size of the graph in check, but new algorithms and layout styles can potentially increase the graph’s legibility. Currently, the edges connected to a vertex are highlighted when a user’s mouse hovers over that vertex. This helps reveal edges that might be obscured by crowding or crossing with other edges, but these connections should be obvious without user interaction. Researchers have developed systems to optimize the appearance of NLDs, particularly when many connections exist between nodes. We can make steps to improve our diagrams by exploring the use of curved edges and minimizing edge crossings, as other works suggest [8].

Most of the interactions within the Relationship Graph view concentrate more on exploring the category terms and not the relationships between them, represented as edges in the graph. We support pivoting on category terms and adding specific category terms to the graph but rarely do we leverage the connections between them for exploration. Since exposing these relationships is a major emphasis of our work, it is only natural that we continue researching ways to use the connections to further users' explorations.

WorkTop was used for a semester by an Italian studies class at Brown. We received valuable feedback from the professor and students of the class and were able to improve the infrastructure and functionality throughout the semester. Unfortunately, most of the features in this paper were not implemented in time to be used by the class. User studies need to be performed to evaluate the efficacy of both search interfaces. Feedback from studies could be used to change features that are not well received or add ones that are deemed necessary.

Conclusion

We introduced two views that highlight interesting trends in search results that would otherwise go unnoticed in traditional search and faceted browsing interfaces. Smart Grouping leads users to documents of interest by using metadata analysis to present the most interesting facets for a dataset. The Relationship Graph view seeks to summarize large document sets by displaying connections between category terms from multiple facets. Unlike existing faceted browsing and search systems, our interfaces minimize the interaction required to find trends and connections within search results. Each view provides tools to improve the layout of data and to leverage the collaborative aspects of the WorkTop system. We feel these visualizations introduce a new way of approaching exploratory search interfaces and see promise for further research in this area in the future.

Acknowledgements

Whenever I used the first-person plural in this paper, I wasn't just employing the "royal we". This work would not have been possible without the guidance of Bob Zeleznik, the Director of Research for the Brown Computer Science Graphics Group and leader of the ongoing WorkTop project. He shared his wealth of knowledge on all things HCI and served as a critic and sounding board to my work. Without his guidance, this work would have looked incredibly different—and not in a good way. Our long discussions will be missed.

Thanks to Andries van Dam for his advising and valuable feedback on my work. He always pushes me to improve as a student and a researcher.

Funding for research on WorkTop was provided by Microsoft Research. Microsoft Bing provided funding for the work in this paper.

The ACM Digital Library provided us with the metadata for SIGCHI conference proceedings from the past 12 years. This dataset made it possible for us to test our visualizations on a much larger scale.

Works Cited

[1] Robert Zeleznik. WorkTop. [Online]. http://www.cs.brown.edu/~bcz/worktop/about.html

[2] Ping Yee, Kirsten Swearingen, Kevin Li, and Marti Hearst, "Faceted Metadata for Image Search and Browsing," vol. Proc. CHI, 2003.

[3] Microsoft PivotViewer. [Online]. http://www.microsoft.com/silverlight/pivotviewer/

[4] Bongshin Lee, Greg Smith, George G. Robertson, Mary Czerwinski, and Desney S. Tan, "FacetLens: exposing trends and relationships to support sensemaking within faceted datasets," in *Proceedings of the 27th international conference on Human factors in computing systems*, 2009.

[5] Greg Smith et al., "FacetMap," *IEEE Transactions on Visualization and Computer Graphics*, vol. 12, no. 5, pp. 794-804, Sept. - Oct. 2006.

[6] Bongshin Lee, Mary Czerwinski, George Robertson, and Benjamin B. Bederson, "Understanding research trends in conferences using PaperLens," in *CHI '05 extended abstracts on Human factors in computing systems*, 2005, pp. 1969--1972.

[7] Marian Dörk, Sheelagh Carpendale, and Carey Williamson, "EdgeMaps: Visualizing Explicit and Implicit Relations," in *Proceedings of VDA 2011: Conference on Visualization and Data Analysis, IS&T/SPIE*, 2011.

[8] Nelson Wong, Sheelagh Carpendale, and Saul Greenberg, "EdgeLens: An Interactive Method for Managing Edge Congestion in Graphs," in *Proceedings of IEEE Symposium on Information Visualization (InfoVis 2003)*, 2003, pp. 51-58.

[9] Graph#. [Online]. http://graphsharp.codeplex.com/