Clarence Chan: clarence@cs.ubc.ca
#63765051

CPSC 533 Proposal
Memoplex++: An augmentation for Memoplex Browser

Introduction

Perusal of text documents and articles is a central process of research in many fields and disciplines. However, there are often vast amounts of literature for any given area of discourse. Thus, the process of extracting useful information from textual sources encompasses not only the review of a large numbers of texts in a general area, but also the filtering and selection of texts relevant to one's research from this multitude of documents. As online electronic document repositories become increasingly prevalent [], it is important that tools be developed to to support their use so that researchers can and extract and filter the information they need from these document repositories. This scenario of finding relevant documents in a large corpus is a specific example of the kinds of issues dealt with in the field of information retrieval.

Some of the more common approaches to information retrieval from a collection of texts are the rather simple concepts of searching and browsing. When a user wants to do a textual **search** of a collection, she will normally interact with the collection through some query interface; the user provides several keywords to the interface, representing a particular query about the area of interest, and is in turn provided with a set of documents that the interface deems to be relevant to the user's initial query. However, while even naïve search algorithms are capable of globally filtering out irrelevant documents when returning results to the user, the list of relevant results returned is often still rather broad. Unless the user knows *exactly* what she is looking for, it becomes difficult to further narrow down the list of results with more specific keyword searches, and the problem of finding the "right" documents becomes a task of **visually** perusing the returned documents, one at a time.

Narrowing It Down: Information Retrieval, Supporting Effective Visual Browsing, Semantic Networks

This makes search less well suited for locating particular items of interest from within a local group of ostensibly relevant results, once the search space has been initially narrowed down. When people begin on research in a given area, they do not always have starting points or specific documents as targets to search for, as they are unfamiliar with the domain anyways. As seen above, the task of further narrowing down an abstract search becomes one of document perusal, or *browsing*. Browsing is the natural, complementary task to search, when the user is in an exploratory state of mind and wishes to peruse a set of documents for potential good hits. In contrast to search, browsing is a less directed task (does not focus on matching a particular query in a large global set), and banks on the user's ability to identify elements of interest by having her **visually** browse through a small local group of related materials.

We can think of browsing in this context as the "second" step of information retrieval following search. Search narrows down a large domain to a smaller set of related materials that are somewhat relevant to one's query; browsing allows a person to further dive into a specific area to identify the key elements of interest. One analogy often used is the process of going to a library, looking at an index of call

numbers to find a specific subject area (search), then going to that section and flipping through the various books in that particular set of stacks to get a feel for the different texts in that area (browsing). At the lower level of browsing through books in the stacks, there is more of an emphasis on local and relational links in the data (article cross-references, related works), rather than the absolute and hierarchical nature of call numbers.

Certain computational constructs have been designed to support similar notions of local and relational links in data; in particular, semantic similarity networks represent sets of items as graphs wherein individual nodes (items) are connected by edges to other nodes based on the strength of their relation to one another. In the case of text documents, semantic networks can be constructed that indicate how closely related different documents in the graph are; documents sharing many keywords, and more importantly, that share many higher-order attributes (as determined by some metric of similarity), are linked together by an edge and can be thought of as tightly related to one another. Because of this, semantic networks are tools that can potentially effectively support browsing of documents.

Joel Lanir's Memoplex Browser is one such interface to a semantic network of documents (the backend of which is Mike Huggett's Ph. D. thesis). The document corpus was a subset of New York Times articles concerning the Iraq war. In the Memoplex Browser, a user can browse a set of documents linked together by a semantic network. Documents are represented as nodes in a radial graph; when a user enters a search, the most relevant node is displayed in the middle of the graph, and its connections to other nodes in the graph are shown. Clicking on other nodes brings them into focus at the center of the graph, and a small portion of the contained text in the document is provided in an adjoining window.

Thus, it is possible to see the nature of the document corpus on-screen, while also supporting the user in finding particular items of interest **visually**. The kind of setup demonstrated in Memoplex Browser is a good step towards using browsing to support information retrieval; however I wish to augment certain features of Memoplex to make it more amenable to local, visual document perusal, and exploratory document navigation.

Domain: Information Retrieval And Visualization
Task: Searching and Browsing
Dataset: Document collections

Problems and Proposed Solution

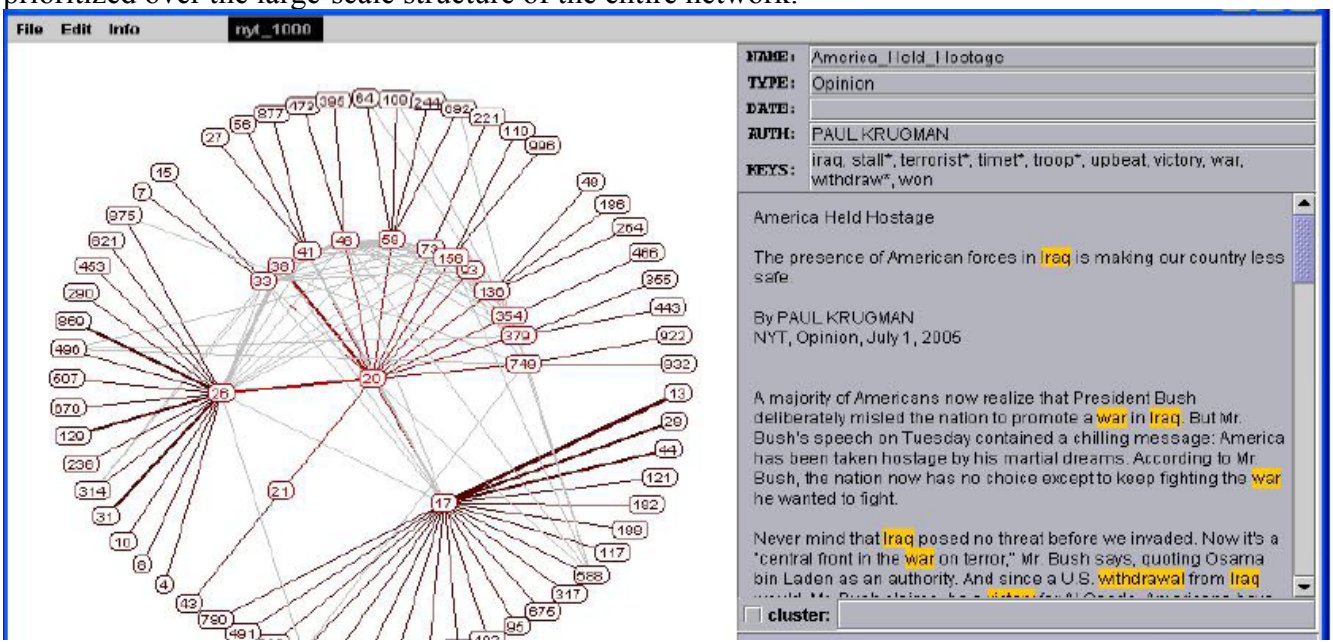
Information Density, Real Estate

One set of issues with the existing Memoplex Browser interface is that only one actual document's text is physically visible at any one time, and more space is devoted to visualizing the entirety of the semantic network than the document of interest itself. It is arguable that being able to visualize the large-scale structure of the semantic network is not as useful as being able to see the actual text contained within those documents, as it is the text that ultimately is evaluated by the user when deciding whether or not a document is useful or not.

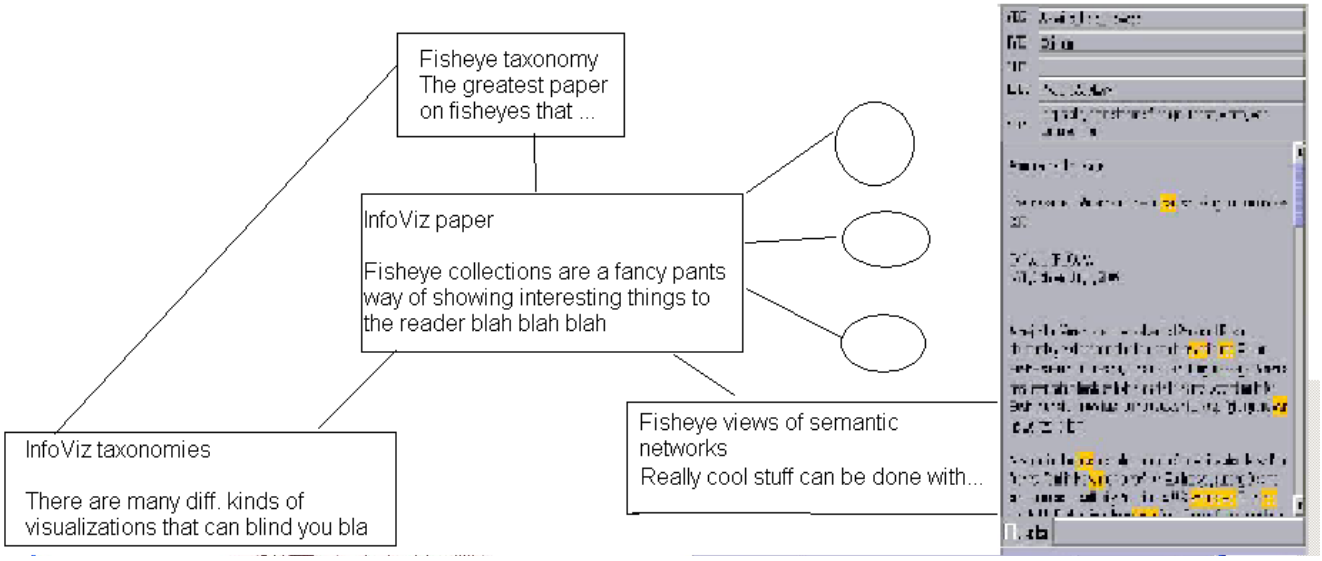
From a task-based point of view, the purpose of the semantic graph should *not* be to visualize the entirety of the network, but to provide connections between the current document of interest and other *related* documents that might be worth browsing and looking at. The rational for using semantic networks in the first place is to support browsing, which is inherently a locally directed task; thus it

stands to reason that only a small number of nodes that are directly connected need to be visualized on the screen at any one time.

Thus, this set of issues concerns the problems of *limited screen real estate*, *information density* and the degree of *scalability* of the visualization (as there is a tradeoff in being able to visualize large-scale structure versus local views of nodes). I propose that the textual information of each document be prioritized over the large-scale structure of the entire network.



Current View Of Memoplex



Modified View. Note that text snippets have been made a part of the nodes, and that much more screen real estate is devoted towards visualizing a few choice nodes of high relevance. Less similar document nodes are still connected but their text is not shown.

As before, the most relevant element to a particular search is made the center of the graph view. However, as seen in the above figures, my proposed solution to this set of problems is to provide a visualization of a much smaller number of nodes with snippets of their actual text embedded in the graph itself. Double clicking on a node brings the full, unabbreviated content of the document into a

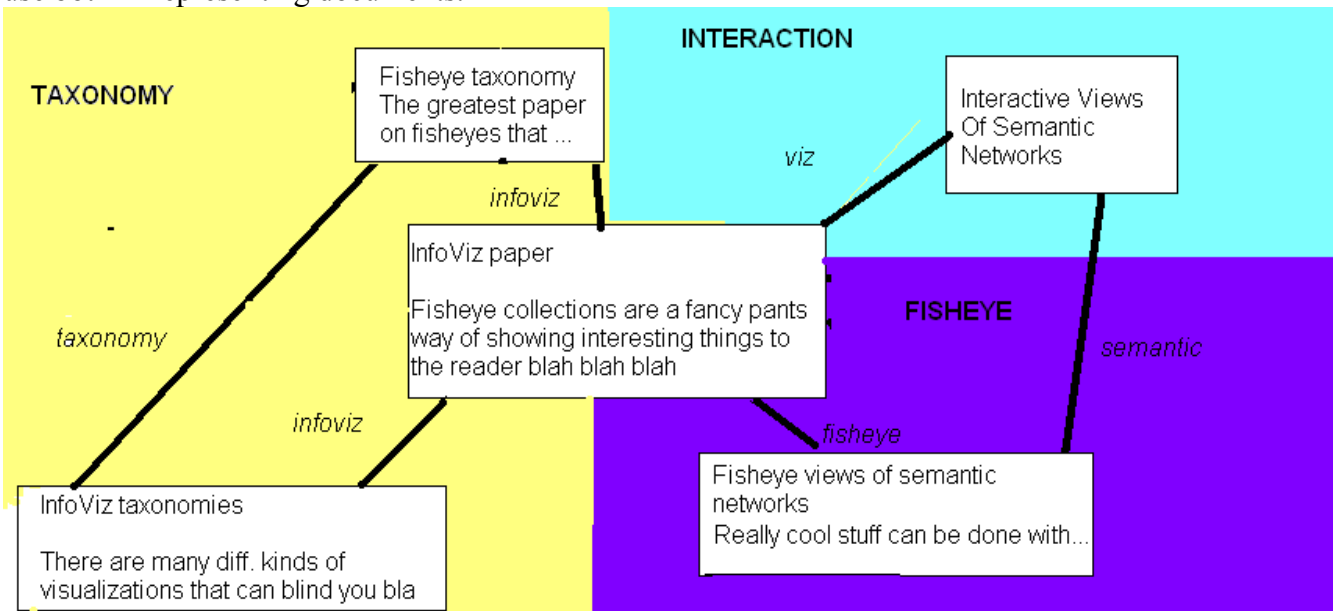
separate window. This would allow for a greater density of relevant information to be communicated to the user through text, which is the meat of the relevant information from the user's point of view. The edges of the semantic network remain to provide links to similar documents "one hop away" from the main document, and those documents are also displayed with their snippets.

It is less relevant to show documents that are two or more hops away as jumping to them does not fit with the browsing paradigm, as users will want to look at local elements that are more closely related first before proceeding to ones further out from the current point of interest. If there are many similar nodes, only the selected strongest nodes will be immediately visible. Clicking on a non-expanded node should "open" it up, closing other nodes if need be to make space. In any case, the original document remains the focus of the view until users click on an expanded adjacent node. As in the original browser, the new node is brought into focus and the current view of the graph centers around that node and its neighbors.

Local Visual Context

I also wish to improve the local visual context provided by Memoplex Browser. The first augmentation I propose is relevant edge labeling. Currrently edges indicate only that two documents are similar, but do not indicate the nature of the similarity. By providing common keywords on edges between items, it may be possible to communicate more accurately the nature of the connection between the two documents so that a user can choose whether or not to view that particular document, thus more efficiently supporting his browsing strategy.

Additionally, I would like to make use of the spatial layout of the local graph view to communicate the different semantic dimensions of the current search space. Currently, spatial location does not encode any attributes in the Memoplex Browser. However, by clustering the documents beforehand, adjacent documents around the current document can be re-arranged around the center using spatial location as an indicator of its strength in each cluster dimension. Currently, the Memoplex Browser can pre-cluster a set of documents, but only uses color and not space to encode this information; I propose to use both in representing documents.



Each of the three colored regions represents a particular keyword dimension; in this case the search is "taxonomy interaction fisheye". Items are placed in their appropriate spatial region. Edges are

labelled in italics.

If at all possible, I would also like to be able to write an algorithm that can take the keywords provided and cluster the corpus according to those keywords, so that the clusters are of more contextual use to the user.

Overall, this set of issues makes for an interesting problem of using space effectively, as well as using appropriate animations to move elements into their approximate spatial locations according to their dimensions when interactively moving between nodes.

Materials
I will be using Prefuse in Java to build these augmentations, together with the existing framework of the Memoplex Browser and the example corpus used by Joel(if I can get ahold of it).

Scenario Of Use

Joe is doing research on interactive fisheye views in visualizations. He has at his disposal an online library of infovis documents that cover many different kinds of visualizations, such as ZUI's, fisheyes, overview+detail interfaces, map visualizations and tree/graph views. He opens up his augmented Memoplex Browser and types 'fisheye visualization interactive'. The Memoplex Browser returns a graph with the most relevant document as suggested by the search engine: Furnas' 1986 paper on fisheye visualizations. He clicks on the document and casually browses through it in the new window, before realizing that it is mostly abstract theory on the concept of fisheyes and degrees of interest, and while he saves this document for use because it is in his general area of interest, it does not have examples of interactive fisheye visualizations and so he looks at the semantic graph to see what else is in the immediate vicinity that might be of use.

Along the 'visualization' axis he sees an infoviz taxonomy paper written by the same author. However, more importantly, long the axis of “interactive” he sees a document on interactive focus+context views. This paper is a little more general than interactive fisheye views, but Joe figures that a lot of related work should fall under the subset of interactive focus+context views that might be fisheye work. He clicks on this document, and it moves to the center, replacing the fisheye paper by Furnas. This new document is now surrounded by two other documents in addition to the Furnas paper which is now classified along the “fisheye” axis most strongly. The other documents are loosely classified in between “visualization” and “interactive”; one of them is a paper on interactive fisheye views, and the other is a paper on interactive map views. Joe double-clicks on the former, finding himself in a possible goal state, and peruses it and finds that it is very useful to his research.

However, Joe is not done. He wants additional examples so he looks at the documents surrounding the interactive fisheye viz paper, which has now moved into the center. He sees two other related items, semantically linked by the graph; he peruses their snippets quickly and saves them for later use as he sees that they seem to be quite relevant based on the abstract alone.

Possible Tarpits, Evaluating Usefulness

There are a number of potential pratfalls in approaching the problems discussed. Conventional approaches to document search, browsing and retrieval have actually been largely successful on a very coarse grained level; the flat lists of absolute relevance results generated by a search engine such as

Google have been proven to be very effective and accurate in what they do. Additionally, the flat list layout provided by search engines allows for more text results to be visualized at once, and sometimes that is all the context that is needed, without requiring links between related documents to be visualized. Even the flat list provided by the "similar pages" option provided by Google is just one click away from looking at similar results that may be of interest to the user.

However, I counter this by arguing that the cost incurred by devoting more space to the representation of the local network and its documents is worth it because the similarity network **directly** brings the locally relevant documents for browsing into view, rather than providing documents of absolute query relevance that may not be related to what the user is looking for. I argue that the provision of context by my proposed visualization outweighs the increased amount of text and results provided by a flat list, as the **locally relevant**, **similar** documents are going to be the ones that the user will more likely be interested in for browsing, than a bulky flat list that contains a sea of results only loosely related to each other.

Another problem lies in using clusters to spatially arrange the documents around the center node. Often, elements related by similarity will also be in the same cluster, thus causing connected documents in a particular view to accumulate moreso in one spatial region of the visualization. This may be a problem but it can possibly be alleviated by forcibly assigning elements to other clusters regardless according to their cluster scores for those particular dimensions, so that their relevance along these other dimensions can also be understood by the user.

All in all, though, it is difficult to affirm these claims without validating them. If time permits I also propose to do a small evaluation of the augmented browser compared to a flat list of results based on the same corpus of documents. Users in the flat list condition would search a list of documents listed in order by absolute relevance with no semantic links, and compare their experiences / satisfaction with that of the augmented Memoplex Browser. This would perhaps be a first step towards determining the exact nature of the tradeoffs between space for more text, and semantic information.

Personal Experience

I have no experience in developing visualizations for document browsing and navigation but I have read some papers on semantic networks and navigation through abstract spaces. I have a bit of experience in Java from undergraduate classes but no GUI programming experience.

Milestones

Week 1 (Oct. 30 – Nov. 5) – Finalize design requirements, discussions with others to iron out kinks, and get ahold of appropriate document corpus
Week 2 (Nov. 6 – Nov. 12) – Get acquainted with Prefuse, and Joel's previous work (basic parsing of documents, building network), Figure out how to draw text boxes atop nodes, f
Week 3 (Nov. 13 – Nov. 19) – Figure out how to rotate text boxes into appropriate spatial dimensions (use predefined clusters first),start work on an alternative list view for user testing
Week 4 (Nov. 20 – Nov. 26) – Make graph dynamically expand the text of nodes when requested by the user. Fix up animation to accommodate, continue work on list view
Week 5 (Nov. 27 – Dec. 3)- Figure out edge labelling and see if clustering algorithm can be written to cluster articles around particular search keywords, start on user testing
Week 6 (Dec. 4 – Dec. 11) – Finish up user testing, write report
Week 7 (Dec. 12 onwards) – Final presentations, huge sigh of relief

References

Hearst, M.A. TileBars: visualization of term distribution information in full text information access, Proceedings of CHI '95, p. 59-66, 1995.

Shneiderman, B., and Marchionini, G. Finding facts vs. Browsing Knowledge in Hypertext systems, Proceedings of Computer, p. 70-80, 1988.

Song, M. Visualization In Information Retrieval: a three-level analysis. Journal of Information Science, p. 3-19, 2000.

Chen, H. et al. Internet Browsing And Searching: User Evaluations of Category Map and Concept Space Techniques. Journal of the American Society For Information Science, p. 582-603, 1998.