Exploring Web-based Visual Interfaces for Searching Research Articles on Digital Library Systems

Maxwell Fowler, Chris Bellis, Christopher Perry, Beomjin Kim
Department of Computer Science
Indiana University-Purdue University
Fort Wayne, IN, U.S.A.
maxfwlr@gmail.com, kimb@ipfw.edu

Abstract
Previous studies that present information archived in digital libraries have used either document meta-data or document content. The current search mechanisms commonly return text-based results that were compiled from the meta-data without reflecting the underlying content. Visual analytics is a possible solution for improving searches by presenting a large amount of information, including document content alongside meta-data, in a limited screen space. This paper introduces a multi-tiered visual interface for searching research articles stored in Digital Library systems. The goals of this system are to allow users to find research papers about their interests in a large work space, to see how document content relates to a search terms, and to refine their search queries using document content. The current, under development pilot system successfully presents graphical illustrations of search results produced from both meta-data and underlying content in an intuitive visual interface that will assist user's search activities. With minor modification, the proposed system can be applied to a variety of other text-based data repositories.

Keywords - Digital libraries; Visualization; Unstructured text content; Visual analytics

Introduction

Academic paper writing leverages online corpora as one of the sources for references to prior work and to build upon previous results. Most corpora are hosted on services aimed at easing the search process; digital libraries such as the ACM Digital Library and the Library of Congress provide books, articles, and other forms of media, while services such as Google Scholar focus on journal papers. While valuable as knowledge repositories, these services lack in their ability to present information in a way that helps lead to easier, more informed decisions when determining which academic papers to read and reference.

Current digital library systems suffer from limiting standards and provide only superficial information in their search results. Most archiving systems display the title of a work, authors, the publication the work appeared in if applicable, and other basic meta-data at first. No profile of underlying document content is provided, which can make finding the best sources a tedious task which requires reading through the plaintext directly. Further, some search systems do not adequately search document content, instead relying upon users to already know the document they wish to retrieve.

The lack of search depth caused by not searching document content is exacerbated by the use of non-intuitive, text based results. This is not an effective form of data representation. Displaying a large amount of text in a column does not provide an efficient way to traverse search results and pinpoint desired content. At best, text based searches can prioritize results on the title that best matches the desired search terms or upon a hidden document relevance score, which does not help a user see why a given paper is the best choice. Further, many text based search systems on digital libraries lack an intuitive way to determine the relationships between titles, the content in documents, and the relationships between different documents.

Visualizations allow data to be presented in manners that are more interconnected and readily processable. This is accomplished by leveraging users' perceptual cognition. Studies have already shown such leveraging leads to faster data consumption and a higher quality of understanding (Card 1999, Veerasamy 1997). Such visualization work has already been applied to some forms of digital libraries in the past. University of Maryland's GRIDL, for example, presents digital libraries using two hierarchical axes with topics on one axis and publication years on another (Shneiderman 2000). The density of documents for that topic and publication year are then displayed as bar graphs, split between the different kinds of digital media in the library. Visualizations, such as GRIDL, allow large quantities of data to be displayed in a coherent format that is tailored for user ease and document content exploration.

Visualization has been used in the past in order to simplify searching document repositories. Most of these visual approaches have used some form of graphical representation to better show links between papers within a document and the overall document spread in a repository.

Some visualization work used a graphing approach with axes. ActiveGraph, developed by Marks et al, used scatter plots with customizable axes (Marks 2005). These axes, the X, Y, and Z axes, could be set to any of the kinds of meta-data discussed earlier. ActiveGraph took a repository wide approach; it did not get into underlying document content, but did allow an at-a-glance look at the entire repository based on specific meta-data.

Others used different graphical representations. Rushall et al. and Lin focused on self-organizing maps that could be directed at a document repository or single book to display the types of documents in a workspace or the topics contained in a repository (Rushall 1996, Lin 1996). These maps were useful for quickly searching for documents in a visual fashion. The search showed the contents of a workspace in a visual form, allowing the user to quickly parse out the kinds of documents provided. This system still lacked a link between the superficial meta-data and document content, though. While preferable to a text search, the work was still plagued by the limiting factor of judging a book by its cover - using the title, but not the actual content within the document.

To address the limitations of only leveraging meta-data and not document content, Short et al. developed a multi-tiered visual interface for digital libraries (Short 2014). This work used the indices of textbooks to index books based upon their overall content and content by chapter. The multiple tiers focused on different representations. The first tier compared books to each other based on desired search terms. This tier was similar to work such as ActiveGraph, leveraging a similar overall interface with a more regimented coordinate system rather than a scatterplot. This allowed for document screening based on meta-data like before. Clearly unrelated titles, works that were too old to be useful, or works with bad reviews could be safely ignored.

Other tiers took a content based approach to visualization. By leveraging the index of textbooks, Short et al. were able to directly allow exploration of document content. The visualizations showed the layout of the book's index and the presence of search terms on a by-chapter basis using the book's index. Searching for topics showed not only books on the subject, but also exposed the relevant content within. This allowed the search to be used to more easily select the best sources, based on how much they covered the desired search topic.

While the work already done is valuable, we see a place for future development. The current visualization work can be applied to other data domains, such as social networking data and unstructured text content. Unstructured content presents a number of issues. Such documents can have different layouts from one another. Even within a specific domain, such as research articles, the structure can be different. While most articles contain similar sections, such as an introduction and a method section, there is no guarantee articles use the same layout. Sacks-Davis and Ron et al. discussed the subject of structuring text content to be indexable and queryable, but did not consider visual approaches or building indexes for journal papers dynamically (Sacks-Davis 1997). Development in this field, utilizing visual analytic techniques, will assist researchers in finding references for their work.

To assist researchers, a visual search focused on research articles in Digital Library systems would be useful. This system requires indices to exist for the content in the papers. These indices need to be searched in a way that will help users make educated decisions on their paper selections. Papers do not tend to have indices, which mandates that an index is built for papers in some fashion in order to be reasonably searched. This is work previously undone, as prior systems that used indices used pre-built ones, such as Greg et al.'s work, and is a topic we need to address.

In addition, a good search term must support the ability for users' queries to undergo search refinement. A search should not only find documents related to given topics, but should allow the user to refine their search using different terms they discovered during the search. Short et al. approached this subject by showing chapter content from books. This is a limitation as current search refinement focuses specifically on content that is searched for. Related content and words that may be synonymous to desired content are currently unexplored angles for search refinement.

We propose a system which will bring the visual aspect and automatic indexing aspect together into one, targeted at assisting researchers in searching text corpora and refining their searches through intermediate results. This paper will introduce an ongoing development of a system; a two-tiered visualization web application that displays research articles with titles and associated content in a graphical format. The first tier will provide a high level profile of the kinds of documents in a repository and how related these documents are to desired search terms. These relationships will also show a relationship between papers, by proxy. The second tier has been designed with the idea of search refinement in mind. It displays the frequency of search terms in the paper, as well as synonyms, terms related to the search terms, and compound terms created by coexisting words. This paper

presents the current prototype of the system developed to assist users' searches on research articles in Digital Library systems.

Methodology

The prototype system consists of three major components: index generation, query processing, and visualization. The index generation module analyzes the underlying content of a Digital Library's research papers and constructs an index for each of them. Query processing is an underlying process that connects the indices with the visualizations. The visualization itself is implemented in two tiers. Tier 1 presents an overview of the document base and the high level relationships among the documents and the query's search terms to guide user' selection of documents. Tier 2 provides a content analysis of a specific document from the Tier 1, showing terms related to the search query for the sake of search refinement. The methodology section is designed around looking for information about thread-based programming and architectures. We used the terms "thread," "process," and "cpu" as our search terms.

Index Generation

Our indexing system was developed using well known Lucene libraries and is not a major focus of our research (Apache 2015b). The documents are first extracted into plain text in order to ensure a consistent format. Using Lucene, common words and other characters deemed to be garbage are removed from the text. This is to prevent such words impeding the index searching process. The text is then stored into a data structure which maintains a word count, as well as information on which sentences in each document contain which words. Together, these structures serve as a searchable index for the document base.

Tier 1 Visualization

The Tier 1 visualization provides a profile of the entire document base. The intent of Tier 1 is to show the best papers for a user's search query in the digital library being used. Tier 1's search is based upon title info and the indices of each document. The aspects considered for each document are the length of the document, the relevancy the documents have to specific terms in a search query, and the relevancy the documents have to the entire search as a whole.

The user's search query is directly represented in the visualization. Each search query is three terms, with each term going into a different colored box. The colored boxes are red, green, and blue, which are the primary additive colors used in computer science. In our examples below, "thread," is the green term, "process," is the red term, and "cpu," is the blue term. Each search term is shown in the visualization in a circle of the term's color. From this point forward, all shapes in the visualization are referred to as nodes.

The documents appear in the visualization as nodes as well. Only documents that include at least one of the aforementioned search terms are placed on the visualization. Node size is determined absolutely, with the largest document in a repository having the largest node and the smallest document the smallest node. Node size is capped at 30 pixels, with any documents that would have a larger size being set to 30. The number of documents displayed by the search is a user defined number, using a slider to change for more tightly focused or broader reaching searches.

The nodes are positioned to show correlation between each document and the search terms in the query. A force directed graph is used for the layout, specifically d3's implementation of Dwyer's algorithm (Bostock 2011, Dwyer 2009). Each document has tension directed towards the search query nodes. The tension force is directly linked to the relation between a document and a term. A search term that a document has no relation to will provide 0 tension. Documents that feature all three terms will tend to be pushed into the middle of the visualization, while documents that only feature two terms will appear between only those two terms and not appear in the middle. We also use a simple collision algorithm to prevent node overlap. The document is placed in the triangle defined by the search term nodes. Documents with all three search terms equally weighted within it will be placed in the middle, equidistantly. Papers more related to a specific term will be placed closer to them, as they have a higher tension toward that search term than the others.

Document relevancy is determined using Lucene's Term Frequency-Inverse Document Frequency (TF-IDF) algorithm. TF-IDF is frequently used in data and text mining applications. The score for a term increases if a term appears often in a document or if that particular word is uncommon (Apache 2015a). Overall, documents are favored for having a large number of desired words. The score for words is used for both document relevancy to a single search term, for the node positioning, and overall document relevancy.

Overall document relevancy combines the relevancy scores for all three search terms to give each document node a color. The most relevant paper, determined by having the highest overall score for all search terms, will be black. Less relevant papers appear white, with papers in the middle falling somewhere on the grayscale in between. Black contrasts well with lighter colored nodes around it, making it a good color to indicate the best papers. Our basis for this decision came from color theory and digital graphics design (Foley 1996).

When a node is selected on the visualization, the node is highlighted and the paper's supplementary data is shown in a tooltip. Figure 2.1 shows a sample of the Tier 1 visualization with the best paper selected. Note that the best paper is not necessarily the largest, as in our sample query one of the smaller papers has the best overall search results. When a paper is selected, the title of the paper is shown, which acts as a link to the PDF. The author and conference are provided as well. Finally, a link to the second tier visualization is provided to link between the two tiers.

The formula scores the probability of a potential term being related to a base term by comparing how many hits the potential and the base have together over the number of hits only the base does in the document set. For each of the two terms that use the score, we will go into specific detail.

Related terms are defined as non-synonyms that appear in the same sentence as a search query's term. These terms can help refine a user's search query by showing them words that commonly appear together.

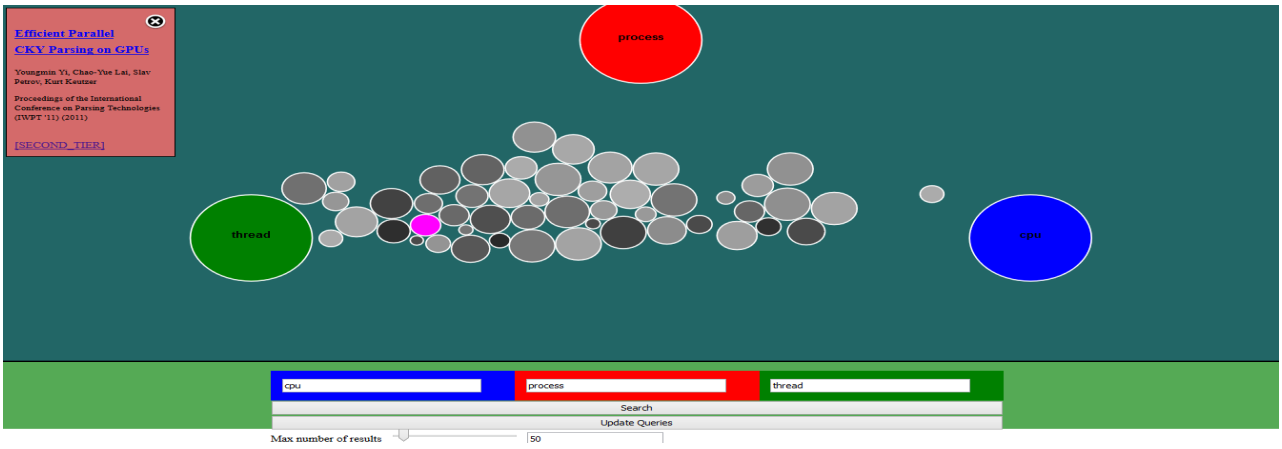

Figure 2.1 Sample view of Tier 1

Tier 2 Visualization

The Tier 2 visualization provides a closer look into specific documents. It relates the search terms to the content in the document itself. This way, the user can see precisely how prevalent a given term is in a document. This serves to increase user confidence in the document they have selected as being a useful document. In addition to directly showing term prevalence, the system provides coexisting words, related terms, and the synonyms for the search terms. The intent is to help users' with the task of search refinement by selecting better words for their queries. Synonyms and compound words are a new consideration in this research. While prior studies did not consider them useful, both serve to allow the user to phrase the same query in multiple ways to find the best results possible for their search.

Both related terms and compound terms using a scoring algorithm called Pointwise Mutual Information-Information Retrieval (PMI-IR). PMI-IR was developed by Peter D. Turney for developing automatic indices of non-structured content (Turney 2001). Our algorithm specifically implements PMI-IR 3, with some modifications:

$$P(potential|base) = \frac{Hits(potential\ AND\ base)}{Hits(base)}$$

This can then be used in a new search to refine the documents returned in a specific direction. In order to determine related terms, all the documents in the database are first stripped down to just contain the sentences containing a specific term.

Each of the remaining words is scored as the potential, with the search term as the base, using PMI-IR 3. The higher the score, the more relevant a specific related term is deemed to be. The current system allows all related terms with a score higher than 0 to appear in the visualization.

Compound terms are similar to related terms, but are specifically terms made up of two words; a query term and either the term directly before or directly after the query term in a sentence. These terms are intended to expand a specific search term. For example, a search can be refined to use "cloud computing," rather than "cloud," after finding the former as a compound term of the latter. Given the query "machine," one might get both, "machine learning," and, "autonomous machine." The same PMI-IR 3 scoring is used on compound terms as is used on related terms.

Synonyms are generated by searching through a synonym database. Our algorithm uses WordNet for collecting synonyms (Fellbaum, C). WordNet returns synsets of potential matches. As synonyms tend to be small in number, there is

no threshold number in place for limiting the number of synonyms displayed.

The visualization is consistent between Tier 1 and Tier 2. The same force graph rules still apply. However, data in this tier is only related to one term node. This means that document content nodes that tend toward the middle are weakly related to their term. Content with a high relatedness to a search term, though, appears close to the term's node. Likewise, node size remains consistent in that it shows size, but the size is the count of specific terms, rather than document size.

The size of each object, including the search terms themselves, are how relevant they are to the overall paper. This is the word count from Lucene's index. It is possible to have a paper where a search term has relevance 0, which would make the shape have 0 size. Likewise, it is possible to have related, compound, or synonym terms be larger than the search query nodes if they appear in the current document more than the query terms do. The largest nodes in Tier 2 are the terms that are most likely to help refine a search by replacing a search term.

Each of the term types is represented with its own node shape. Synonyms are given circular nodes, to show they are directly related in meaning to the search query terms. Related terms and compound terms, meanwhile, are squares and triangles respectively. This decision was made to draw distinction between term types.

Figure 2.2 shows an example of a Tier 2 visualization, specifically from the last figure's best document. The size of the three search term nodes shows that they are, in fact, all three prevalent in the paper. Thread is the most relevant, though, as shown by the size. We can see what the terms are by hovering over their nodes. The term will appear in a tooltip above the node.

Discussion

The proposed system has made ground towards reaching the goals set for it. The visualization successfully functions on unstructured text content, such as the journal papers used in this study, which has yet to be done in this way. The Tier 1 visualization does provide a visually accessible look at the entire document repository. It manages to capture the legibility of previous systems while improving upon the visualization's ability to aid in selecting documents. Further, the Tier 2 visualization does make strides towards helping users refine their search in meaningful ways.

The Tier 1 visualization is quite strong at this point. It is useful for finding papers that span across multiple related domains, as shown in the methodology section. The overview is scalable, allowing users to search for a large number of papers or select only a small subset refined to be the best for a given search. Further, the white to black color scale for least to most relevant allows the most relevant paper to stand out easily, making finding the best options in any sized search an easy task.

Tier 1's strength is obvious when we compare the visual search to a text based alternative. Figure 3.1 shows a search for the terms "simulate", "transform", and "automata". The search provides the same information the visualization does, but the best paper's relation to terms is shown as numeric scores. This is less intuitive than the visualization's black to white color scale and position algorithm.

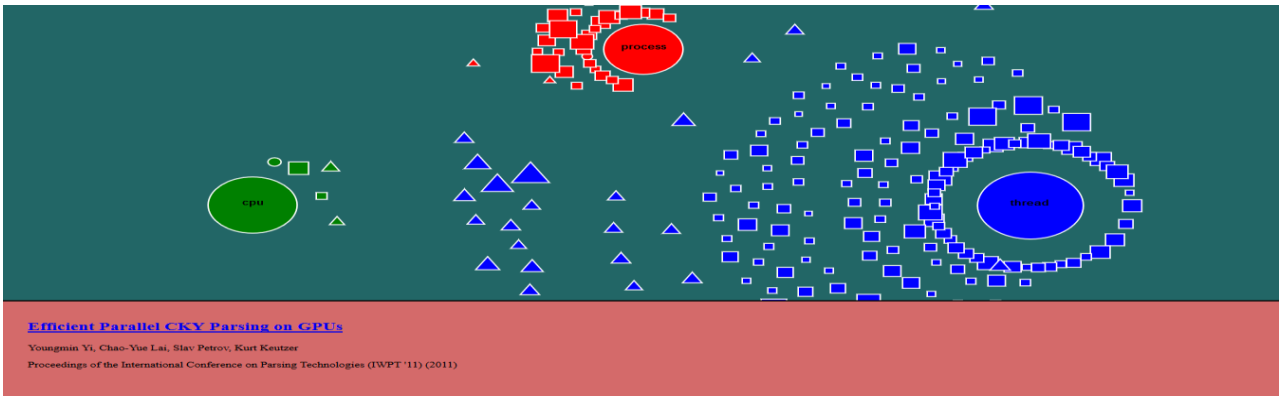

Figure 2.2 Sample view of Tier 2

Figure 3.1 Sample view of text based search designed for usability tests

Figure 3.2 shows two searches. The search on the left is the same as the search in figure 3.1. It becomes readily apparent where the two best papers are and how they related to the terms. It also becomes apparent that the world simulate is fairly useless. Using the Tier 2 visualization, we refined the search to use, "grammar," rather than, "simulate," giving us the image on the right. This search refinement gives documents of much higher quality, according to the color scale and provides a better distribution in the visualization's center. Further, the refinement is easier to make in the visual system than a text based one, which would require reading the whole document to find useful terms.

The Tier 2 visualization succeeds in the goal of providing potential search refinement. It shows all the potentially useful related terms each of the search results have. The figure above directly shows the benefit of search refinement, as previously discussed.

The choice of red, green, and blue for the search term nodes was retained for Tier 2 in order to allow RGB color combinations to show off terms related to multiple documents. This was abandoned in practice, in part because meaningful terms related to two distinct, other terms were rare. This means the colorization here could be changed to represent different information if a better way to show term relation is found. Further, some collision can occur in tier 2 term nodes, which needs addressed in future updates. This can be seen in Figure 3.3, especially around the term "thread".

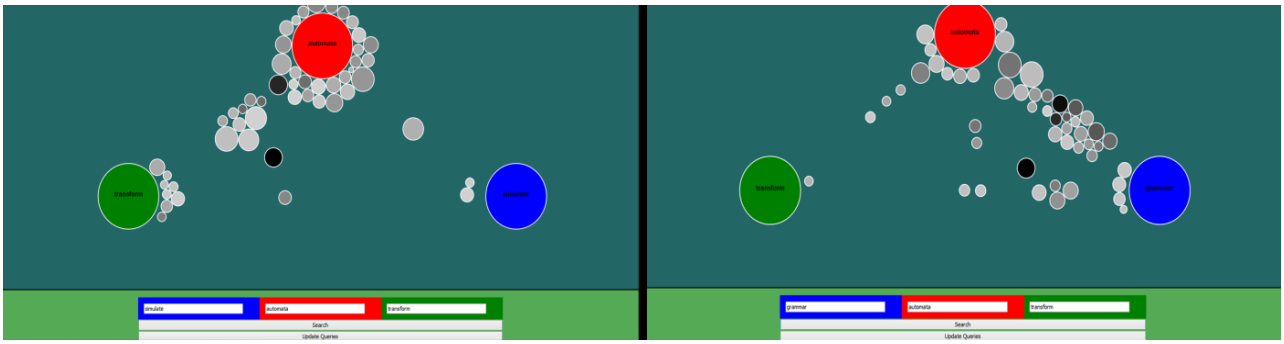

Figure 3.2 The visual form of Figure 3.1 and a refined version

Figure 3.3 The best paper from 3.2 shows some Tier 2 overlap

FUTURE WORK

The proposed system makes good strides at reaching the goals set out in the introduction. Despite this, there are future work avenues to prove the system works, improve the system, and potentially apply the system in other ways. Some discussion of those angles follows below.

The paper uses searches on a paper database of our own creation. It is built of papers freely accessible on Google's own research papers. There are approximately 1600 of them. In the future, we will apply our visualization to a paid-for paper collection, such as the Text Retrieval Conference Proceedings (TREC). This will give our system a more robust collection to be tested against.

Usability tests are needed to prove that the visualization above is better than a text based system. While we feel the visualization system stands on its own merit, usability tests will add credence to that claim. Our next task will use a custom made, text based search system and compare it to our visualization. We will use a metric based approach to judge effectiveness, as well as judge user preferences. This way, the visualization's greater effectiveness compared to traditional text based interfaces can be proven.

The visualization is not without room for improvement. It would be ideal to take into account more than just the presence of search terms in Tier 1. Ideally, the document ranking algorithm can be altered to take into account all attributes of a document. This means the document's titles, reviews, word count, presence of search terms, and other data will all contribute to a document's relevance score.

Currently, the search is related to three terms. This is left over from earlier work when we were considering using RGB color combinations for paper quality, rather than the current black and white scale. We retained the color usage for Tier 2, but did not find such situations that would benefit from color combination. We are considering allowing an arbitrary N-gon, which will free up RGB colors to be used for different visual elements. Such an N-gon's size will be determined by the user, which a minimum size of two to allow the visualization to remain fully featured.

The potential exists for a third tier: specific term expansion. This tier would allow the user to select a term and see more information about it, including all related terms to that term in the repository, synonyms both in the repository and outside of it, and definitional information. This has yet to be implemented as the usefulness is questionable. It may be sufficient to augment the Tier 2 visualization with word definitions and leave it at that.

Another room for improvement is the clustering algorithm, especially when it comes to the overlapping related terms and the large clusters of low use papers. Some form of blobbing algorithm which combines closely related papers into one node which can then be expanded into the full node set should be considered to make the visualization simpler and more user friendly in such instances.

Finally, future work could include applying our system to other domains. So long as an index can be constructed for the desired data, any form of text-based data could be searched and visualized using the above system. For example, social network posts could be used as documents to search. This would allow the system to search blogs discussions forums, and other forms of social media for the sake of determining user consensus or gathering data for marketing purposes.

CONCLUSION

This paper proposed a visual search on Digital Library systems, specifically targeting journal papers and other research publications. The proposed system targets two goals. The system's first goal is a visualization on an entire document base, to help the user more easily see the best papers available for a given search. The second goal is to aid in search refinement, changing the original search to better suit the user's needs. This unexplored element was addressed by providing a visualization for various forms of related terms. A preliminary indexing step allowed us to apply these visual elements to a collection of unstructured text data; while not our primary research focus, this was still an interesting element and is in contrast to prior visualizations of structured document content which did not require an indexing step.

We developed a two tier system to meet our goals. Tier 1 provides a visualization over the document base for a specific set of three search terms. The papers are positioned and colored based on their relevance to terms and the overall search respectively. Tier 2 provides a visualization of a document's content. It shows how the search terms relate to the underlying content and show other related terms for the sake of search refinement.

By providing these two tiers, we help users with multiple tasks. Tier 1 makes it easy to see if a given search is useful or if a given search is skewed too far to one term. Tier 1 also makes it easy to find the best papers for a given search. Tier 2 allows us to confirm the best paper shown includes the search terms with a high frequency. Tier 2 also lets us refine our searches, allowing users to turn bad searches into good searches by changing a search term or two.

By assisting users with these tasks, our system makes sufficient strides towards our goals. Our last step is fixing the

problems mentioned in the discussion section and investigating the improvements mentioned in the future work section. Once this is accomplished, our system will become practical and serve users in their searching of unstructured content in digital libraries.

References

Apache Software Foundation (2015). *Class TFIDFSimilarity*, Available: https://lucene.apache.org/core/5_2_1/core/org/apache/lucene/search/similarities/TFIDFSimilarity.html

Apache Software Foundation (2015). *Lucene 5.2.1 core API*, Available: https://lucene.apache.org/core/5_2_1/core/overview-summary.html#overview_description

Bostock, M., Ogievetsky, V., Heer, J. (2011). *D3: Data-Driven Documents* in *IEEE Trans. Visualization & Comp. Graphics*.

Card, S. K., Mackinlay, J. D., and Shneiderman, B. (1999). *Information visualization* in *Readings in Information Visualization: Using Vision to Think*, pp. 1-34.

Dwyer, T. (2009). *Scalable, Versatile and Simple Constrained Graph Layout* in *IEEE-VGTC Symposium on Visualization*.

Fellbaum, C. (2005). *What is WordNet*. Princeton University.

Foley, J., van Dam, A., Feiner, S., Hughes, J. (1996). Computer Graphics: Principles and Practice, Addison-Wesley Publishing Company.

Lin, X. (1996). *Graphical table of contents* in *Proc. of the first ACM Int. Conf. on Digital Libraries*, pp. 45-53.

Marks, L., McMahon T., and Luce, R. (2005). *ActiveGraph: a digital library visualization tool* in *International Journal on Digital Libraries*, vol. 5, no. 1, pp. 57-69.

Rushall, D., and Ilgen, M. (1996). *A context vector-based self organizing map for information visualization* in *TIPSTER: Proc. of a Workshop on held at Vienna, Virginia*, pp. 159-166.

Sacks-Davis, R., Dao, T., Thom, J. A., Zobel J. (1997). *Indexing documents for queries on structure, content and attributes* in *Proc. of International Symposium on Digital Media Information Base (DMIB)*.

Shneiderman, B., Feldman, D., and Rose, A. (2000). *Visualizing Digital Library Search Results with Categorical and Hierarchical Axes* in *Proc. 5th ACM International Conference on Digital Libraries*, pp. 57-66.

Short, G., and Kim, B. (2014). *Multi-tiered Visual Interfaces for Book Search with Digital Library Systems* in *Proceedings of the 6th International Conference on Multimedia, Computer Graphics and Broadcasting*, pp.21-24.

Turney, P. D. (2001). *Mining the Web for Synonyms: PMI-IR versus LSA on TOEFL* in *Proc. of the 12th European Conference on Machine Learning (EMCL '01)*, pp. 491-502.

Veerasamy, A., and Heikes, R. (1997). *Effectiveness of a graphical display of retrieval results* in *Proc. of the 20th Annu. Int. ACM SIGIR Conf. on Research and Development of Information Retrieval*, pp. 236-245.