Term Distribution Visualizations with a Focus+Context Model

Moses Schwartz, Curtis Hash, and L.M. Liebrock, *Member, IEEE*

Abstract—Many text searches are meant to identify one particular fact or one particular section of a document. Unfortunately, predominant search paradigms focus mostly on identifying relevant documents and leave the burden of within-document searching on the user. This research explores term distribution visualizations as a means to more clearly identify both the relevance of documents and the location of specific information within them. We present a set of term distribution visualizations and introduce a Focus+Context model for within-document search and navigation.

Index Terms—Document visualization, term distribution, Focus+Context, information retrieval.

1 Introduction

Many text searches are meant to identify one particular fact or one particular section of a document. For example, users referencing a manual seek to quickly learn how to perform a task; digital forensic analysts seek to find specific artifacts that may be used as evidence of wrongdoing. Unfortunately, predominant information retrieval paradigms do not emphasize this sort of within-document search. Here, the primary emphasis of the search is not to simply find relevant documents, but to identify specific sections within those documents. This field of research, especially with regard to information visualization for full-text and within-document information retrieval, has not received enough attention from researchers [15].

Early information access systems focused primarily on searching titles and abstracts to identify relevant documents [9]. This paradigm has not changed significantly, even as technology has advanced and full-text documents have become the norm. Although search engines have access to full-text and can better identify relevant documents, common search technologies do not take full advantage of the presence of a full-text logical document view. As described in Section 2, there have been numerous efforts to create within-document search aids, but none have been widely deployed.

Visualizations of search results are an obvious venue for improving usability in both between- and within-document search applications. Unfortunately, the very nature of language and the difficulties of natural language processing render it difficult to design effective visualizations [1]. The most common approach to surmounting this problem has been to examine the structure and distribution of terms within a document [3, 7, 9, 12]. Visualizations of structure and term distribution can aid the user in identifying relevant documents and relevant sections within those documents. In essence, this supports comparing the relative value of different documents and different sections within documents.

In this paper, we present a set of term distribution visualizations building on prior work in within-document searching, propose a model for within-document searching with these visualizations, discuss the additions of a Focus+Context model for navigation and variable-granularity searches, and enumerate several fields in which our visualizations might be applied. The primary contributions of this work are:

- Exploration of extensions to the TileBars [9] and Relevance Curves [11] visualizations;
- Application of TileBar-like visualizations as a primary navigation and search aid; and
- Introduction of a Focus+Context model to term distribution visualizations for variable-granularity searches.

We discuss preliminary studies on implementation details such as color and hue selection and blending. Furthermore, we introduce possible applications to elaborate on the potential utility of our design. Implementation of a distributable interface and extensive user studies for both visualization design and applications are currently under way.

In our visualization model, we create a sequential histogram of query terms throughout a document, and present this information as one of our set of visualization variants. The Focus+Context model consists of a brushed section of the visualization expanded into a new, full-size visualization with finer granularity. Figure 1 provides a simplistic example of a term distribution visualization and the Focus+Context model. This particular example is visualizing a plain-text version of Lewis Carrol's *Through the Looking Glass* [4], using "Alice," "Humpty," "Tweedledum," and "Tweedledee" as search terms.

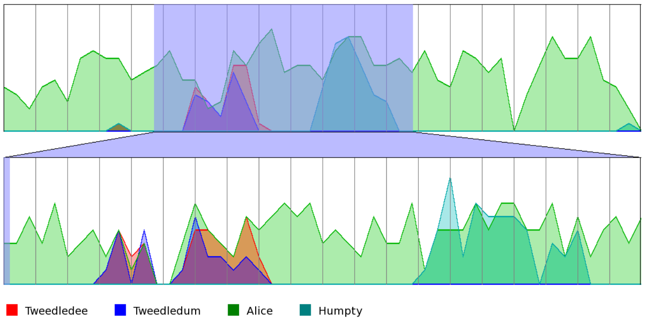


Fig. 1. An example of our visualizations and Focus+Context model visualizing a plain-text version of Lewis Carrol's *Through the Looking Glass* [4], using "Alice," "Humpty," "Tweedledum," and "Tweedledee" as search terms. An expanded version of this image is revisited in Section 3.6.

Section 2 of this paper describes related work. Section 3 presents our visualizations and Focus+Context model. Section 4 discusses usability aspects of this research. Section 5 explores several potential applications of the visualizations, such as potential use in digital forensic string searches. We present plans for future work in Section 6, and Section 7 provides concluding remarks.

2 Related Work

The focus of most information retrieval research has traditionally been to return a list of ranked documents, as one routinely sees in modern search engines. Helping the user to search and navigate within

- *Moses Schwartz is with the Department of Computer Science, New Mexico Institute of Mining and Technology, E-mail: moses@nmt.edu.*
- *Curtis Hash is with the Department of Computer Science, New Mexico Institute of Mining and Technology, E-mail: chash@nmt.edu.*
- *L.M. Liebrock is with the Department of Computer Science, New Mexico Institute of Mining and Technology, E-mail: liebrock@cs.nmt.edu.*

Manuscript received DD MMMM YYYY; accepted DD MMMM YYYY; posted online DD MMMM YYYY.
For information on obtaining reprints of this article, please send e-mail to: tvcg@computer.org.

the document is a somewhat less popular, but very interesting field of information retrieval; despite the decreased popularity, there has been considerable work on within-document searching. Because our focus is on the visualization rather than specific information retrieval aspects, this discussion of related work focuses on visualizations only, and neglects work on text categorization and search methods that do not have significant visual components.

TileBars are an early influential visualization for providing relevance feedback and aiding within-document searching [9]. The TileBars method takes a set of search terms and creates a matrix of tiles, each row representing the entire document, each column representing a block of text in the document, and the darkness of the tile representing the frequency of a search term in the block. See Figure 2 for an example, which illustrates some of the power of this technique. The final document in the figure never has the two search terms appearing near each other; that document is less likely to be of interest than the first document.

TileBars were intended to compactly indicate relative document length, query term frequency, and query term distribution to assist a user in assessing whether a document is relevant for the given search terms and to identify relevant sections or passages. Although TileBars have not been widely adopted, the concept remains elegant and useful, and the original TileBars work has been cited by many papers on within-document searching. Our visualizations are built upon the concept of TileBars and our implementation includes a visualization similar to the original TileBars.

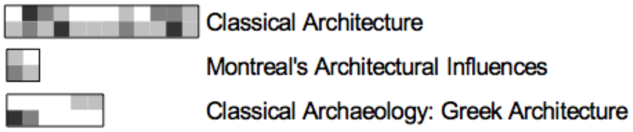


Fig. 2. TileBars visualizing three documents. Search terms are “classical” (in the top row) and “architecture” (in the bottom row). Image from [11].

“Visualization of WWW-Search Results” [11] and an accompanying case study [10] present a system utilizing several visualizations—scatterplots, bargraphs, TileBars, relevance curves (see Figure 3), and thumbnail views—to aid in searches of the world wide web. The majority of the visualizations are for identifying relevant documents rather than within-document searching, but many of the principles applied can be extended to our project. The concept of integrating a suite of disparate but complementary visualizations into a within-document search tool appears viable and useful. Note specifically that our histogram visualizations (see Section 3) are, essentially, an extension of the relevance curves visualization.

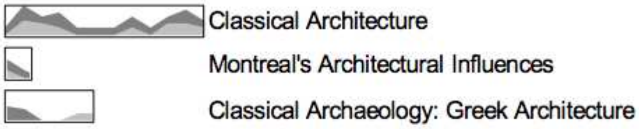


Fig. 3. Relevance curves visualizing three documents. Search terms are “classical” and “architecture”. Image from [11].

“A Scrollbar-based Visualization For Document Navigation” [3] describes a visualization system using a TileBars-like concept to indicate the location of search terms within a text file. The system highlights search terms in a document and places small icons of corresponding color in the vertical scrollbar, enabling a user to quickly scroll to relevant sections. User studies have shown that users respond well to this subtle search aid and the addition of this technique to our own interface is a natural extension. This work is relevant to ours, but does not attempt to act as a primary search and navigation aid.

The Spoken Content-based Audio Navigation (SCAN) [13] user interface addresses the same issue of within-document searching that we are interested in, but with the additional focus of searching speech archives. SCAN utilizes automatic speech recognition to obtain a partial transcript of speech recordings, then performs searches very similar to our own utilizing a straightforward histogram to indicate the relevant sections of a recording. SCAN does not, however, provide granular information about the occurrences of each term within a document and does not provide a mechanism for brushing and drilling down.

ProfileSkim (originally presented as SmartSkim in [7]) addresses the same within-document searching problem that we are researching and has an interface that is very similar to ours. User studies for ProfileSkim have been very positive, indicating that this sort of within-document searching technique is useful and valuable [6]. ProfileSkim creates a histogram of a document showing only a calculated relevance score for each section, on the assumption that the cognitive load on a user would be excessive with a visualization more like TileBars. However, while empirical studies would be required to make any strong conclusions, ProfileSkim does not appear suitable for tasks other than typical document search and navigation. Specifically, ProfileSkim does not implement any sort of brushing, Focus+Context, or zooming interface for dealing with large files. Further, ProfileSkim's relevance scores make relationships between term frequencies difficult to discern.

Full-text visualizations have been discussed in the context of data mining, as in [14]. These data mining approaches to full-text analysis identify patterns and relationships within textual corpora. However, the focus of data mining research is different from straightforward information retrieval—data mining techniques might be used to identify relevant terms that could then be searched for in our model.

“Sequential Document Visualization” [12] is one of the most recent works that is similar to ours. The research takes a largely mathematical approach to the problem of within-document searching by identifying patterns within the text and fitting the frequencies to a curve. The Interactive Document Visualization Toolkit presents users with several types of visualizations built on the statistical models. An informal user study showed largely positive results, although some of the advanced visualizations were ranked poorly because (it is surmised) they are relatively unintuitive and the subjects had little experience or time for training. While this work is relevant and may be complementary to our research, there is not much overlap in approaches.

3 Visualization Techniques

We present two types of visualizations, TileBars and histograms, for use as part of a Query-Browse (QB) information retrieval model [1, 15]. For each visualization type, the distribution of terms may be measured using either a sliding window or blocks. Both visualizations may be used in grayscale or color and. both support search queries of arbitrary length.

All examples in this section have been generated on a plain-text version of Lewis Carrol's *Through the Looking Glass* [4], using “Alice,” “Humpty,” “Tweedledum,” and “Tweedledee” as search terms. Throughout this section, the reader may note that some visualizations are better than the others at displaying a particular type of information—for example, grayscale histograms excel at showing overlap, but do not provide useful information on each individual term. Initial analysis of the effectiveness and shortcomings of each visualization are presented; future work will validate these conclusions with a larger user study.

3.1 Calculating Distributions

Throughout this study, for both TileBars and histograms, term distributions are used for both blocked and sliding window cases. These distributions are calculated very simply, but the use of the terms *blocked* and *sliding window* must be defined for this context. For blocked distributions, we split a file into chunks of some arbitrary number of words and calculate the raw frequency of each search term within each of those chunks. For the sliding window distributions, we perform the same calculation, but rather than calculating search term frequencies within each chunk, frequencies are calculated within a sliding window.

For clarity, consider the following mathematical explanation, which holds for both blocked and sliding window distributions. For each search term i, the set $F^i = \{f_n \in F^i \mid f_n = C_{i,n}/S_w \text{ for } 0 \le n \le L_d - (S_w - S_i)/S_i \text{ and } S_i \le S_w\}$ is generated, where $C_{i,n}$ is the number of occurences of search term i within the n^{th} block, S_w is the size of the window, S_i is the size of the sliding window increment, and L_d is the length of the document. When $S_i = S_w$, the distribution is blocked; otherwise, it is a sliding window distribution.

With $S_i = 1$, the distribution is as continuous as possible on a discrete dataset, but this is also a processor-intensive calculation; experience has shown that setting S_i to a reasonable fraction of S_w (in the neighborhood of $S_i = S_w/5$) will result in effective visualizations. All of the sliding window visualizations shown in this paper were generated with such a setting.

3.2 TileBars

The variants of TileBars presented in this paper are extensions to the initial concept. As in the original, the visualizations are all essentially matrices of tiles, with darkness and color blending representing the frequency of search terms.

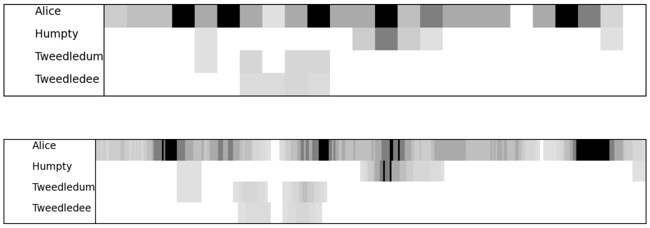


Fig. 4. Blocked (top) and sliding window (bottom) Classic TileBars visualizing Lewis Carrol's *Through the Looking Glass* [4] with search terms "Alice," "Humpty," "Tweedledum," and "Tweedledee."

3.2.1 Classic TileBars

Our simplest visualization emulates the original TileBars by calculating term frequencies over discrete blocks, using one term per row, with grayscale intensity representing frequencies. This visualization may be the most intuitive and easiest to read and is useful for identifying sections of a document with term overlap. However, this visualization's presentation of information is coarse in comparison to our other visualization variants and it becomes harder to read with many rows.

Sliding window TileBars are functionally identical to classic TileBars, but show more subtle changes in term distribution across the document. It is not yet clear whether the finer-grained information is useful when presented in this fashion. Figure 4 shows an example of these two variants of TileBars. Note that the two visualizations, despite visualizing the same data set, have significant differences. In particular the larger granularity in the blocked TileBars has an entire block of a document showing high concentration for a term, whereas the sliding window has much smaller slices. Therefore, if there is high concentration of a term in a small area, it is shown more accurately in the sliding window. However, a similar result is possible by using more (smaller) blocks in the blocked approach. Also note that the apparent misalignment of high (or low) concentrations in the visualizations is due to the combination of granularity being used and the specific location of terms relative to the block boundaries. This "misalignment" is evident in Figure 4, where we can see that some of the high-frequency blocks for "Alice" and "Humpty" appear to be in different locations in the blocked and sliding window variants. The differences between sliding window and blocked visualizations are further explored in Figure 7.

3.2.2 Color TileBars

Color TileBars display two terms, each in a different color, on each row of the TileBar and show term overlap through color blending. The design decision of using red and blue for the colors is arbitrary, and may change depending on user study results. Attempting to use color blending to convey information introduces many difficulties and decreases usability, as discussed in [5]. However, color TileBars display the same information in half the rows that grayscale TileBars do and therefore may be useful in space-constrained environments.

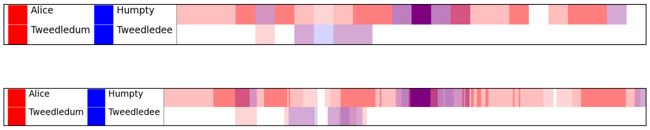


Fig. 5. Blocked (top) and sliding window (bottom) Color TileBars visualizing Lewis Carrol's *Through the Looking Glass* [4] with search terms "Alice," "Humpty," "Tweedledum," and "Tweedledee."

Analogous to the earlier case, sliding window color TileBars are functionally equivalent to color TileBars, but use a sliding window frequency distribution. As with sliding window grayscale TileBars, it is clear that more information about the distribution is presented, but the utility of that information remains to be fully tested. Figure 5 shows an example of the results from our implementations.

Further experiments with this approach will explore use of hashing and color weaving instead of color blending. We will consider whether it how much cognitive effort is required to understand color blending versus the the use of different types of hash marks (say diagonal and counter diagonal) for two terms. Further, our user studies will explore the limits of the number of search terms versus the use of greyscale or color. Since color (or mixed hashing) reduces the space to visually explore, it may be a better approach when many search terms are being considered together.

3.3 Histograms

The histograms presented in this paper are an extension to the TileBars concept, although the information displayed is quite different from classic TileBars. Relevance curves, as described in [11], are very similar to our visualization, but our histograms can display more terms with significantly finer granularity. Furthermore, while our color TileBars extension is limited to displaying the frequencies of two search terms per row, the histograms are capable of displaying several sets of search term frequencies on the same graph.

As with the TileBars visualization, there are numerous other extensions to analyze in order to determine what is most effective for users. The use of hashing and color versus space, as well as the use of multiple histograms and how to visualize many terms using this approach will be explored in future work.

3.3.1 Greyscale Histograms

Greyscale histograms visualize term frequencies on a sequential histogram, without the use of color blending. Each search term is displayed in greyscale on the sequential histogram; overlapping segments are darker. Each horizontal section represents a block of the document, similar to TileBar columns. Histograms offer an additional metric of frequency by displaying frequency values as peaks on the graph. A scale is provided on the left of the graph for reference that indicates the number of occurrences found in a block. Figure 6 shows an example of the results of our implementations.

As with the sliding window TileBar variants, sliding window histograms are functionally equivalent to their blocked counterparts. However, an interesting feature of our sliding window histograms is that the first derivative of the curve (i.e., the slope) reflects the rate of change within the distribution; decreasing slopes indicate decreasing frequency and increasing slopes indicate increasing frequency. In contrast, the slope in blocked histograms represents only a very coarse trend from block to block. Again, note that the sliding window visualization has some immediately evident differences from the blocked variant. Figure 7 shows and elaborates on an example.

While greyscale histograms are not particularly useful for determining what terms are located in what parts of the document, the heavily shaded region resulting from multiple layers of overlapping

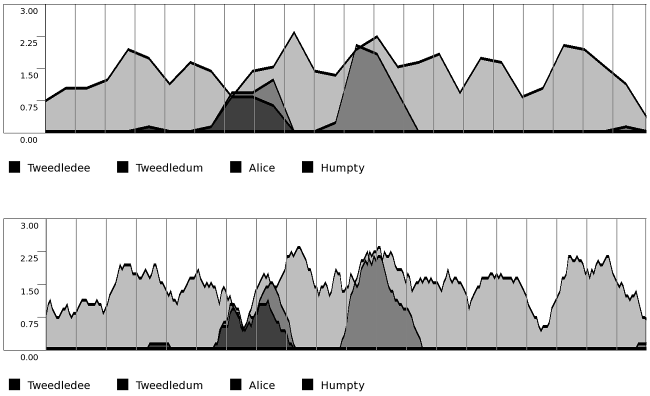
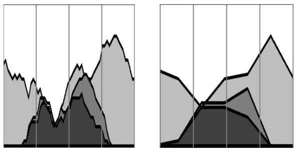


Fig. 6. Blocked (top) and sliding window (bottom) grayscale histograms visualizing Lewis Carrol's *Through the Looking Glass* [4] with search terms “Alice,” “Humpty,” “Tweedledum,” and “Tweedledee.”

histograms is both easier to distinguish—in comparision to the color blended sections of color histograms—and indicative of the close proximity of several terms within the region.

Whether the additional information about term frequency available in the sliding window variant is warranted when one cannot differentiate terms from each other in the visualization is a point worth exploring, but it may allow overlap to be more precisely identified.



Sliding window

Blocked

Fig. 7. A section of the blocked histogram compared to the same block from the sliding window histogram. Note the dip in frequency in the sliding window variant, where the more coarse blocked visualization shows none. This is caused by the presence of a poem in the text that contains none of the search terms. Because the poem is much shorter than the blocks, the blocked version does not show this dip at all.

3.3.2 Color Histograms

Color histograms show overlap through color blending, but appear to remain usable with fewer than four terms overlapping [5]. The addition of a solid, unblended line along the top of the curve allows one to see details that may be obscured by blending. The sliding window variant of color histograms appears much more interesting than its grayscale counterpart, as it is possible to discern trends and fluctuations for each term very precisely.

Usability of the color blending is one of our primary concerns with this visualization, but—assuming that the information may be easily discerned, whether through the current color blending or future attempts with color weaving—the very granular information available and the intuitive nature of the visualization are promising. Figure 8 shows an example of the results of our implementations.

3.3.3 Color Line Histograms

Color line histograms (which are the same as color histograms without fill underneath the line) are particularly interesting to us as they possess most of the advantages of the color-blending histograms, without the downsides of color blending. We expect that these visualizations will be most applicable to tasks in which the user is very interested in the precise frequencies of terms and is willing to put forth more cognitive effort to discern overlap and frequency. Figure 9 shows an example of the results of our implementations.

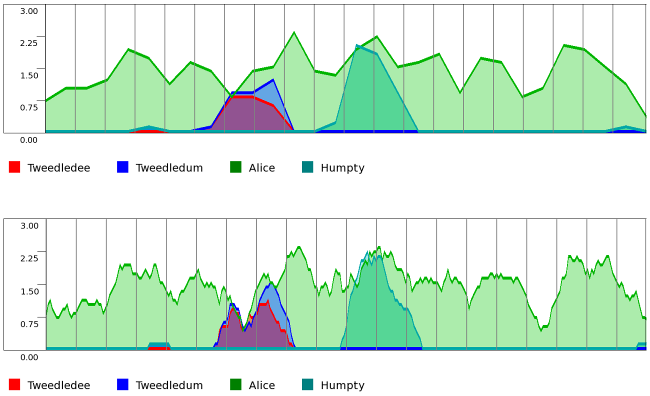


Fig. 8. Blocked (top) and sliding window (bottom) color histograms visualizing Lewis Carroll's *Through the Looking Glass* [4] with search terms “Alice,” “Humpty,” “Tweedledum,” and “Tweedledee.”

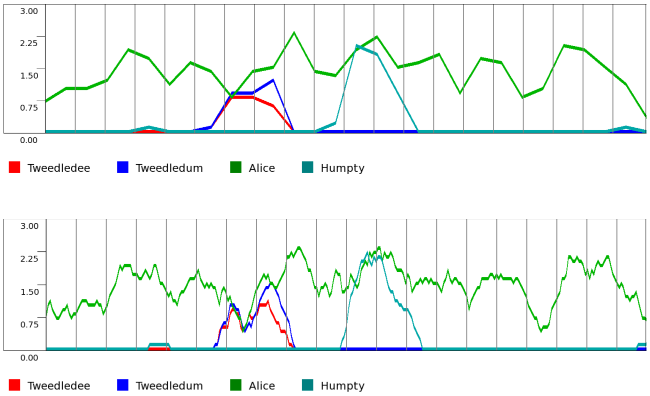


Fig. 9. Blocked (top) and sliding window (bottom) color line histograms visualizing Lewis Carroll's *Through the Looking Glass* [4] with search terms “Alice,” “Humpty,” “Tweedledum,” and “Tweedledee.”

These visualizations require less cognition to interpret blending, but it may be slightly less obvious where multiple terms occur. Future explorations will include combining visualization techniques to show term concentration and show term frequency. For example, one approach would be to use color blending , color weaving, or hashing techniques in a TileBar across the bottom of the graph to quickly show where terms occur simultaneously and use the line histogram to show frequency for each term.

3.4 Focus+Context

A primary contribution of this research is the addition of a Focus+Context model to the straightforward planar visualizations. By brushing an area of interest within the TileBar or Histogram, a user is able to focus on more fine-grained information about the text being visualized, while the context mechanism provides a “big picture” view and maintains the user's sense of locality within the overall dataset.

Figure 10 shows a notional example of the Focus+Context model in use. The brushed section of the initial visualization is re-visualized with a finer granularity; the original visualization, with the brushed section highlighted, remains visible to provide contextual information. The precise presentation of the visualizations within this Focus+Context model may be considered an implementation detail, with many opportunities to increase usability.

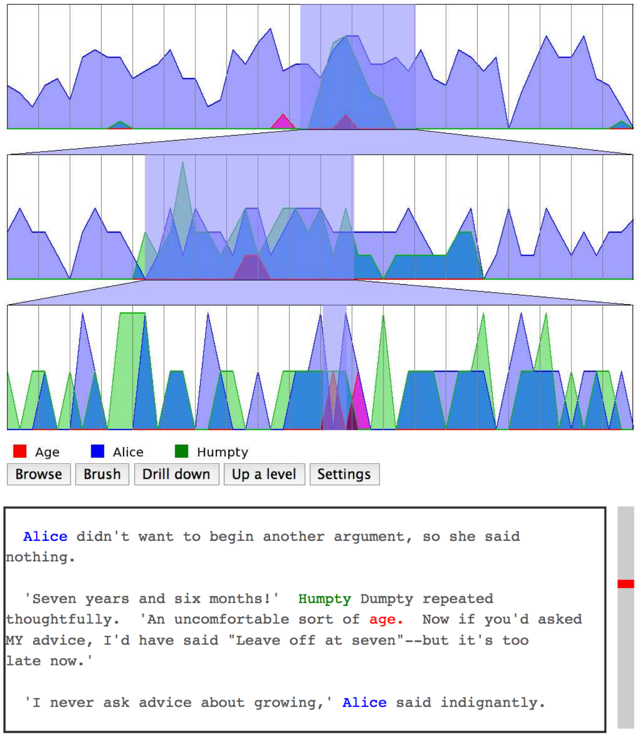
As described in the next section, the Focus+Context model is designed to be intimately tied to the display of the actual contents of the dataset. Thus, while the high-level visualizations provide an overall view of the dataset, the Focus+Context mechanism provides a method to link sections, phrases, or even individual words directly to the visualization.

3.5 User Interface Design

Though more complex than a typical search interface like those in Internet search engines, the graphical user interface is very straightforward. The user specifies a textual dataset (one or more text files) to search and provides a search query consisting of one or more words. The interface backend generates the visualizations using word frequency statistics across the supplied dataset. The visualizations may intially be displayed as thumbnails; clicking on a thumbnail allows the user to navigate and interact with a particular visualization in one panel, while the text of the selected document is displayed in another panel. In addition to Focus+Context, the features of the interface include search term highlighting, zoom, and the ability to click on a location within the visualization and display the corresponding text.

3.6 Use-Case Walkthrough

To elaborate on the interaction paradigm, consider a use-case and a quick walk-through. Suppose one is visualizing *Through the Looking Glass* [4] (as has been shown throughout this paper) and wants to retrieve one piece of information: Alice's age. (Admittedly, this is not difficult to find with conventional search methods, nor is Carroll's work the object of many information retrieval tasks, but it serves as an entertaining example). There are in fact two instances of Alice stating her age, so let us focus on the one that occurred during a conversation with Humpty Dumpty. A user would first specify search terms, the document to search, and the type of visualization to use. In this case, the search terms are "Alice", "Humpty", and "Age", and we will use the aesthetically-pleasing blocked color histograms. An initial visualization is then presented (the top graph in Figure 10) and the user can inspect overall trends throughout the document. The user may elect to brush a section and drill down to view finer-grained information. As can be seen in the image, it is easy to identify the two sections that use the word Age. We brush the section containing references to Humpty and Age and drill down, generating the middle graph. We then brush and drill once more, to obtain a more granular view. The user then may slide a selection window across the drilled-down visualization; the text box below the visualization contains the text corresponding to the selected section of text. The relevant text may be identified by color, which matches those colors used in the visualization. As can be seen in Figure 10, Alice is "Seven years and six months" old.



Alice didn't want to begin another argument, so she said nothing.

'Seven years and six months!' Humpty Dumpty repeated thoughtfully. 'An uncomfortable sort of age. Now if you'd asked MY advice, I'd have said "Leave off at seven"--but it's too late now.'

'I never ask advice about growing,' Alice said indignantly.

Fig. 10. Our interface showing the use of Focus+Context to identify a particular section of *Through the Looking Glass* [4]: Alice telling Humpty Dumpty how old she is. We brush and drill-down twice to get to a highly granular view, then move a selection window across the lowermost visualization. The text in the box below corresponds directly to the selected part of the visualization.

3.7 Complexity and Performance Considerations

The presented visualizations are extremely simple and fast to generate and should function well even on low-end desktop systems. However, the calculation of term frequency distributions can be processor and I/O intensive, especially in the case of large files or sliding window distributions with small window increments (e.g., $S_i = 1$). This may be alleviated through the use of more advanced file indexing algorithms or by performing these CPU-intensive tasks on a separate high-performance parallel backend.

4 Usability

Informal, qualitative data solicited from colleagues suggest that the visualizations and Focus+Context model are intuitive and may be helpful in information retrieval tasks. Early work on TileBars demonstrated that term distribution visualizations were useful relevance-feedback mechanisms [9]. User studies for more recent work, such as [7], also suggest that this type of visualization and interaction paradigm are effective for certain tasks.

We have taken preliminary steps toward empirically testing the validity and effectiveness of this design. We performed an initial user study and attempted to take quantitative measures by timing subjects while they attempted to identify specific sections in documents using our visualizations. However, we encountered technical difficulties with both the implementation and the design of the user study and did not get statistically significant results. Despite that, useful qualitative feedback was obtained. Classic TileBars and grayscale histograms received the most positive responses (as might be expected, given the intuitive nature of overlap and darkness in grayscale). However, some users found that the color-blended visualizations were more useful when information about each individual term was needed. The technical difficulties mentioned above are being resolved, resulting in a better user interface, visualization extensions, and a better understanding of issues for a future user study.

5 Applications

Although no individual component of the research presented in this paper is groundbreaking, the combination of techniques into a new search and navigation paradigm suggests numerous potential applications, a few of which are introduced below.

5.1 Generic Text Search

The most straightforward application of these visualizations is that of conventional searches of textual documents. This application is very similar to the original purpose of [7], [9], and other work as described in Section 2. The visualizations could be used to complement traditional search paradigms in the same fashion as [9], or as a standalone document navigation aid, as in [7].

5.2 Term Distribution Morphology

Our visualizations are also well-suited for the analysis of term distribution morphology; in other words, the visualizations provide information about how the frequency of a particular set of terms changes throughout a dataset. For example, one might use the contents of a community-based website, such as Slashdot [1], as a dataset and attempt to visualize the rise and fall of Internet memes such as "In Soviet Russia...".[2] Similar work was done in [8]. This concept has also been discussed in the context of data mining [14].

[1] http://slashdot.org

[2] http://www.urbandictionary.com/define.php?term=in+soviet+russia

5.3 Digital Forensics

Information retrieval is an integral part of the forensic analysis of digital media. Many current forensic toolkits—for example, Sleuthkit[3]—utilize the Unix command line utility Grep to search files for relevant information. Even EnCase [4], a leading commercial forensic suite, only offers a Grep-like text search tool. This shifts much of the work onto the user, as she must wade through considerable amounts of mostly irrelevant material returned by conventional search utilities.

More advanced string searching for digital forensics has been discussed in works such as [2] and visualizations have been identified as one method of improving string searching. Our visualizations can easily be applied as a search aid for digital forensics, as digital forensic string searching is simply a specialized extension of general text searching.

Other applications for digital forensics include visualizations of log files, visualizations of file content based on access and modification time (using a timestamp rather than offset within a file for the visualizations' horizontal axes), or even visualization of text extracted from network traffic.

6 Future Work

The work described in this paper is an initial exploration into the feasibility of applying our term distribution visualizations to the information retrieval field. Several opportunities for future work immediately present themselves. After implementation of some planned extensions, the critical work will focus on formal user studies to be performed to validate the utility of the visualizations, as well as to indicate aspects that warrant additional research. Likely stemming from such studies, the visualizations themselves may be further extended and improved. Finally, the visualizations are being implemented in a distributable software package to be used in real-world applications.

6.1 Usability Testing

As the visualizations have not been formally tested, despite apparent face and concurrent validity, user testing is warranted prior to a full implementation. As such, work on additional usability testing is currently underway.

Tests will include basic usability, but also tests that compare the various visualizations for use in different settings. For example, are hashing or color mixing more effective when there are many search terms? We will also make direct comparisons to alternative text searching methods such as Grep.

6.2 Visualization Extensions

Term distributions like those presented in this paper are easily modified, as is evident from our own extensions and related work. It is likely user studies will reveal ways that the visualizations presented here can be extended to be more effective for conveying information.

One major topic for future work on the visualizations is improved color blending and the incorporation of color weaving techniques, as discussed in [5]. In addition, the use of hashing and weaving will be compared with color blending to determine effectiveness. Finally, as noted earlier, there are opportunities to combine different techniques to separate some of the information (overlapping concentration versus specific term frequency). This may provide an opportunity to reduce the cognitive load associated with color mixing.

6.3 Case Studies

After initial formal testing, application of the visualizations to real-world problems will be pursued. This will first entail the creation of production-ready software, customized to help accomplish specific tasks. Using this software in real world applications will lend further credibility to the visualizations and interaction paradigm, as well as introducing more opportunities for improvement.

7 Conclusions

Although further user studies must be conducted to determine the efficacy of the visualizations, initial research shows considerable promise. Our approach to Focus+Context provides the ability to "drill-down" within a large document and identify specific information, a feature lacking in related work. Term distribution visualizations have many applications and further research will be beneficial both to the information retrieval research community and to the many fields that employ advanced information retrieval methods.

Acknowledgements

This work was supported in part by NSF Grant #0313885.

References

[1] R. A. Baeza-Yates and B. A. Ribeiro-Neto. *Modern Information Retrieval*. ACM Press / Addison-Wesley, 1999.

[2] N. L. Beebe and J. G. Clark. Digital forensic text string searching: Improving information retrieval effectiveness by thematically clustering search results. In *Digital Investigation*, volume 4 supplement 1, September 2007.

[3] D. Byrd. A scrollbar-based visualization for document navigation. In *Proceedings of the Fourth ACM International Conference on Digital Libraries*, 1999.

[4] L. Carroll. *Through the Looking Glass*. Project Gutenberg, 1991.

[5] H. Hagh-Shenas, S. Kim, V. Interrante, and C. Healey. Weaving versus blending: a quantitative assessment of the information carrying capacities of two alternative methods for conveying multivariate data with color. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1270–1277, Nov.-Dec. 2007.

[6] D. Harper, I. Koychev, Y. Sun, and I. Pirie. Within-document retrieval: A user-centred evaluation of relevance profiling. In *Information Retrieval, 7, 265–290*, 2004.

[7] D. J. Harper, S. Coulthard, and S. Yixing. A language modelling approach to relevance profiling for document browsing. In *JCDL '02: Proceedings of the 2nd ACM/IEEE-CS joint conference on Digital Libraries*, New York, NY, USA, 2002.

[8] S. Havre, E. Hetzler, P. Whitney, and L. Nowell. ThemeRiver: Visualizing thematic changes in large document collections. *IEEE Transactions on Visualization and Computer Graphics*, 8(1):9–20, 2002.

[9] M. A. Hearst. Tilebars: visualization of term distribution information in full text information access. In *CHI '95: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 59–66, New York, NY, USA, 1995. ACM Press/Addison-Wesley Publishing Co.

[10] T. Mann and H. Reiterer. Case study: A combined visualization approach for www-search results.

[11] T. M. Mann. Visualization of WWW-search results. In *DEXA Workshop*, pages 264–268, 1999.

[12] Y. Mao, J. V. Dillon, and G. Lebanon. Sequential document visualization. In *IEEE Transactions on Visualization and Computer Graphics*, volume 13, No. 6, pages 1208–1215, November/December 2007.

[13] S. Whittaker, J. Hirschberg, J. Choi, D. Hindle, F. C. N. Pereira, and A. Singhal. SCAN: Designing and evaluating user interfaces to support retrieval from speech archives. In *Research and Development in Information Retrieval*, pages 26–33, 1999.

[14] P. C. Wong, W. Cowley, H. Foote, E. Jurrus, and J. Thomas. Visualizing sequential patterns for text mining. In *INFOVIS '00: Proceedings of the IEEE Symposium on Information Vizualization 2000*, page 105, 2000.

[15] J. Zhang. *Visualization for Information Retrieval*. Springer, 1st edition, December 2007.

[3] http://www.sleuthkit.org

[4] http://www.guidancesoftware.com