

Evaluating a Visual Presentation of Retrieved Documents

Anton Leuski
Center for Intelligent Information Retrieval
Department of Computer Science
University of Massachusetts
Amherst, MA 01003 USA
leuski@cs.umass.edu

Abstract

In this paper we study a visualization system for the purpose of helping the user to locate interesting material in the retrieved data. The system works by placing the documents into 1-, 2-, and 3-dimensional space and positioning them according to the inter-document similarity. We compute the quality of the visualization by simulating a user searching for the relevant material and calculating average precision of that search. We compare the numbers for the visualization with the same measure taken for traditional ranked list. We show that the visualization performs – on average – significantly better than ranked list. We show a significant advantage of multidimensional visualizations over the 1 dimensional one. We also show that the difference between 2 and 3 dimensional visualizations is very small.

1 Introduction and Related Work

An information retrieval system places retrieved documents in a list in the order they are most likely to be relevant: the first document is the best match to the user’s query, the second is the next most likely to be helpful, and so on. The user of the system is expected to follow the system recommendations – start from the top of the ranked list and follow it down looking at the documents one by one. In the ideal case the user will see all the relevant documents before any non-relevant ones. We are interested in situation when this simple model breaks down and the relevant documents appear to be scattered all over the ranked list. We study alternative ways of organizing and browsing the retrieved documents that might help the user to locate the interesting information quickly without needing to wade through a lot of non-relevant material.

In this paper we conduct an experimental analysis of an information visualization system that works by placing documents in 1-, 2-, or 3-dimensional space and positioning them according to inter-document similarity. We present significant support for the use of such a system as an alternative to a ranked list for browsing retrieval results. We show that with the visualization relevant documents can be found more quickly.

1.1 Related Work

Multiple visualization approaches have been developed in recent years. Generally these visualizations are designed to present some type of patterns in a document set and they are considered to be browsing interfaces. The format of the presentation varies significantly from system to system. For example, Hearst et al. [13] suggest a clustering system that groups the retrieved documents into five (or another preselected number) clusters and displays them simultaneously as lists of titles. A similar presentation was developed by Leuski and Croft [15], however they do not limit the number of clusters and their display looks more like the traditional ranked list.

It is very common for the information organization to be presented graphically. The documents, paragraphs, and concepts are usually shown as points or objects in space with their relative position indicating how closely they are related. Allan [1, 2] developed a visualization for showing the relationship between

documents and parts of documents. It arrayed the documents around an oval and connected them when their similarity was strong enough.

The Vibe system [9] is a 2-D display that shows how documents relate to each other in terms of user-selected dimensions. The documents being browsed are placed in the center of a circle. The user can locate any number of terms inside the circle and along its edge, where they form “gravity wells” that attract documents depending on the significance of those terms in that document. The user can shift the location of terms and adjust their weights to better understand the relationships between the documents.

High-powered graphics workstations and the visual appeal of 3-dimensional graphics have encouraged efforts to present document relationships in 3-space. The LyberWorld system [14] includes an implementation of the Vibe system described above, but presented in 3-space. The user still must select terms, but now the terms are placed on the surface of a sphere rather than the edge of a circle. The additional dimension should allow the user to see separation more readily.

Our system is similar in approach to the Bead system [6] in that both use forms of spring embedding for placing high-dimensional objects in 3-space. The Bead research did not investigate the question of separating relevant and non-relevant documents. The system was designed to handle very small documents – bibliographic records represented by human-assigned keywords. Leuski and Allan [16] extend the Bead spring-embedding approach and applied it to complete, full-sized documents. They introduce the notion of visualization quality for the retrieval purposes and conduct an extensive study of how the dimensionality of visualization affects its quality. They also suggest two methods for incorporating user feedback into the visualization.

The number of efforts to develop graphical information visualizations suggests their perceived importance, but there have been few efforts to evaluate their quality. In this study, we focus on evaluating the visualization for a specific task. We will show that the visualization results in substantial improvement in a hypothetical user’s ability to find relevant material rapidly. In the next section we will describe the evaluation methodology we used to conduct our experiments. We then proceed by describing the experiments and the results. We conclude with discussion of future work.

2 Evaluation Methodology

Leuski and Allan [16] established an evaluation methodology for estimating quality of interactive information organization systems. The essence of their approach is to select a particular task for the system and the user, and then replace the user with an automatic simulated strategy that searches for the documents in the organizational structure created by the system. They propose this approach as an inexpensive laboratory way to obtain preliminary evaluation of a system’s performance before committing to a user study. In this paper we follow their evaluation framework. We begin by establishing the experimental task for the analysis, we then describe the system design, the search strategies, and performance measure.

2.1 Experimental Task

The task of locating the relevant information is the process we analyze in this study. We assume that the user has located a few of the relevant documents – we believe this is a reasonable strategy and almost always could be done by looking at the titles in the ranked list. We investigate how the visualization helps to locate the rest of the interesting documents. Thus, the experimental task is defined: Given that some of the documents presented by the information organization system are marked as relevant or non-relevant, isolate the rest of the relevant material.

2.2 System Design

For this study we adopt the vector-space model of documents. Each document is represented by a vector V such that v_i is a *tf idf* weight of the *i*th term in the vocabulary:

$$v_i = \frac{tf}{tf + 0.5 + 1.5 \frac{doclen}{avgdoclen}} \cdot \frac{\log(\frac{N+0.5}{docf})}{\log(N + 1)} \quad (1)$$

where tf is the number of times the term occurs in the document, $docf$ is the number of documents the term occurs in, and N is the number of documents in the collection. The distance between a pair of documents is measured by the *sine* of the angle between the corresponding vectors. This measure of distance is widely used in vector-space model [18].

We want to visualize the vectors in 1-, 2-, or 3-dimensions preserving the relative distances (or relative similarities) among them. For this purpose we employ a multidimensional scaling approach that is called spring-embedding. Spring-embedding is a force directed placement graph drawing algorithm that generates an approximate solution to a graph layout when the distances between connected nodes are given as constraints [10]. The constraints are modeled as springs. The inter-document similarities serve as constraints in our experiments.

It has been pointed out [16] that if all the constraints are incorporated into the spring-embedding process the resulting structure is very tight and resembles a “soccer-ball”. To prevent this from happening a threshold parameter is introduced into the algorithm – all constraints that fall below a predefined value are removed from the process. It allows us to generate more “interesting” spatial structures. Unfortunately, we now have a parameter that we do not know how to select.

Our intuition is that the interesting embeddings will have some amount of spatial structure – e.g., they will exhibit clumps. This intuition is partially grounded in the Cluster Hypothesis: “closely associated documents tend to be relevant to the same requests” [19, p.45]. Croft [8] showed that this hypothesis holds in a retrieved set. Thus, the relevant documents should be similar to each other and therefore should show up in the visualization as a clump. So we choose to search for clumps. To measure the “clumpiness” we turn to the point field theory [7]. There a statistic called the *K-function* is introduced [5] that estimates the average tightness of a point pattern:

$$K(h) = \lambda^{-1} E(N(h)), \quad h \geq 0, \quad (2)$$

where $E(\cdot)$ is the expectation operator on the point set and λ is the “intensity” of the point set. In other words, the *K-function* is the average number of points in the point set within distance h of any point in this set, normalized by the mean number of points in a unit volume of space. The values of the *K-function* are then compared to the values of the *K-function* for a known random pattern – i.e., a pattern that does not exhibit any clumps. The amount of difference between the two statistics is used to judge the clumpiness of the original point pattern. This approach is described in greater detail elsewhere [17].

2.3 Search Strategy

We measure the quality of the visualization by defining a search strategy that simulates a user looking for the documents in a point pattern representing the document set. We have experimented with multiple search strategies. Two of the strategies are presented in this paper:

Single Document Strategy We assume that we know at least one relevant document. The strategy starts at an arbitrary known relevant document and proceeds by analyzing the rest of the unknown documents in proximity order relative to the starting point. If there are several possible starting points (we know more than one relevant document) the final performance is averaged across all starting points.

Cluster Centroid Strategy We assume that we know relevance judgments for some of the documents. The strategy begins by defining a cluster that contains all the known relevant documents. It then proceeds by analyzing the rest of the unknown documents in proximity order relative to the center of the cluster. If the document is relevant, it is added to the cluster and the centroid shifts.

2.4 Performance Measure

Our search strategies consider the unknown documents in a particular order that depends on the relative spatial location of the documents and other factors. This order of the documents is considered to be another ranked list. Given the order of the documents we evaluate the ranking by calculating non-interpolated average precision [12]. When there are multiple possible orderings, we use the average of all of them.

3 Experiments

In this section we describe the testbed that we used for running our experiments. We then proceed by describing seven experimental questions that we consider, presenting their results, and analyzing the implications of our results where appropriate. The questions we consider are:

1. Can we choose a threshold for spring embedding?
2. Are more dimensions better for the embedding approach?
3. What is the benefit of different search strategies?
4. How does the embedding compare to the classical ranked list?

3.1 Experimental testbed

For our experiments we used TREC [11] ad-hoc queries with their corresponding collections and relevance judgments. Specifically, TREC topics 251-300 were converted into queries and run against the documents in TREC volumes 2 and 4 (2.1GB). For each TREC topic we considered four types of queries: (1) a query constructed by extensive analysis and expansion [3]; (2) the description field of the topic; (3) the title of the topic; and (4) a query constructed from the title by expanding it using Local Context Analysis (LCA) [20].

In addition, we used TREC topics 301-350 to create queries to be run against TREC volumes 4 and 5 (2.2GB). Again, the same four different types of queries were constructed, except instead of just using the description field for the second query type, we used both the title and the description field of the topic. The description fields for topics 301-350 were constructed to assume the presence of the title. In contrast to an earlier study [16], where the authors considered only queries which produced a reasonable amount of relevant documents in the top 10 documents, we do not limit our query sets.

For each query we selected the 50 highest ranked documents and embedded them in 1, 2, and 3 dimensions. To “seed” our search strategies with a starting point we assumed that the relevance judgments for all the documents from the top of the ranked list to the first relevant document are available to the system. This is similar to a situation when a user starts reading documents from the top of the ranked list and continues until reaching a relevant document. However, we did not use this information to construct embeddings. The earlier study [16] suggests an approach to incorporate relevance information into the visualization. The ranked list was treated as an embedding in 1 dimension where the documents are positioned according to their rank values. Note, that both search strategies considered in this study will traverse the ranked list in the traditional way – from the top (or from the highest ranked unknown document) to the bottom. Thus, the precision scores should correspond to the precision scores that are usually reported for an information retrieval system.

3.2 Threshold Selection

Recall the threshold that avoids tight soccer-ball configurations. For N documents there are generally $N(N - 1)/2$ different threshold values – for 50 documents that results in 1225 different embeddings. What is the expected quality of the visualization when we are forced to select the threshold for the embedding randomly? Can we improve on that by considering only embeddings with high spatial structure – i.e., “clumpy” embeddings?

We measure the spatial structure in a point pattern by considering the difference between the K -function computed for the point pattern and the K -function computed for a known unstructured pattern. For the latter we used a Poisson generated pattern. In contrast with the earlier study [16] where the authors studied only the absolute difference in the K -functions, six parameters were considered: minimum difference, maximum difference, absolute value of the difference, and normalized versions of each parameter among embeddings for each individual query. We then performed a linear regression of normalized values of precision on these six parameters across 50 TREC-5 full queries embedded in 2 dimensions. We chose the normalized precision over the actual precision for two reasons. First, we use the linear model generated by the regression to predict relative values of precision as compared to the exact numbers – we need the model to select the

Table 1: Threshold selection procedure effect. The first 1D, 2D, and 3D columns are for the case when no threshold selection procedure is applied. The second 1D, 2D, and 3D columns is for the case when the thresholds are selected using a regression model. Percent improvement is from without threshold selection to with threshold selection for that dimensionality.

Query sets			Embedding					
			w/o threshold selection			w/ threshold selection		
id			1D	2D	3D	1D	2D	3D
1	TREC-5	Full	29.3	31.3	31.6	28.4 (-3.0 %)	38.1 (+21.5*%)	39.9 (+26.2*%)
2		Desc	26.4	30.4	31.3	23.6 (-10.7*%)	44.2 (+45.4*%)	46.2 (+47.6*%)
3		Title	22.2	25.3	25.8	22.0 (-0.9 %)	36.8 (+45.3*%)	38.1 (+47.4*%)
4		Title + Desc	23.6	25.7	26.0	22.2 (-6.0* %)	35.3 (+37.2*%)	37.5 (+43.9*%)
5	TREC-6	Full	35.2	39.8	40.8	33.6 (-4.7* %)	50.3 (+26.3*%)	51.4 (+26.1*%)
6		Desc	31.6	36.8	38.1	30.7 (-2.9 %)	51.4 (+39.5*%)	52.8 (+38.3*%)
7		Title	29.7	35.1	36.5	29.6 (-0.3 %)	51.1 (+45.3*%)	52.7 (+44.6*%)
8		Title + Desc	33.4	39.0	39.9	32.7 (-2.1 %)	50.0 (+28.1*%)	52.2 (+30.6*%)
averaged on 2-8			28.9	33.2	34.1	27.8 (-3.9* %)	45.6 (+37.3*%)	47.3 (+38.7*%)
averaged on 5-8			32.5	37.7	38.8	31.6 (-2.6* %)	50.7 (+34.4*%)	52.3 (+34.6*%)

best embedding among a set of all possible ones for a given query. Second, we also observed that the model that regresses normalized precision has a slightly better data fit than the one that tries to predict the actual precision values.

The resulting regression model was then considered as the threshold selection algorithm – we select the embedding that produces the highest value for the linear model. We then studied how the threshold selection improves our chances of selecting an embedding with a good precision value. We also trained different linear regression models on both 1 and 3 dimensional data. Generally the results were consistent with what we observed for the 2 dimensional model. We do not report those numbers.

In this section we provide the results we obtained in our experiments. Each table shows the corresponding average precision numbers and the percentage improvement over the baseline data. A star in the percentage column designates that the change in the precision is statistically significant according to t-test with significance level $p < 0.05$. We report the values for both TREC-5 and TREC-6 query sets in 4 different modifications. Recall that the threshold selection linear regression model was trained on the embeddings for TREC-5 full queries. The rest of the queries for TREC-5 (title, descriptions, and titles with descriptions) are different and therefore result in different document sets and might have been considered for testing. However, we draw our conclusions only on TREC-6 queries; the rest of the data is provided to illustrate the robustness of the procedure – that the effect does not change much when switching from training to testing data.

Table 1 shows the effect our threshold selection procedure has on the embedding quality. Cluster Centroid search strategy was employed for these experiments. We provide the results for all 8 different query sets and across 3 different dimensions for embedding. The threshold selection significantly improves the expected value of the embedding in both 2 and 3 dimensions by 34.4% and 34.6% respectively. It does hurt in the one dimensional case – we observe a slightly worse embedding than we could get by randomly selecting one.

3.3 Dimensionality Effect

Despite intuition that a high dimensional visualization provides more degrees of freedom and therefore a better chance to represent the inter-document relationship accurately, the earlier study [16] found that there is almost no difference in the quality of the embeddings in 2- and 3- dimensions. We were interested in repeating this study with a larger data set and a better threshold selection procedure.

Table 2 shows a clear advantage of multidimensional embeddings over the 1-dimensional ones. We observe

Table 2: Dimensionality effect. Visualization quality evaluation of different query sets in different dimensions. Percent improvement is from 1D to 2D, from 1D to 3D, and from 2D to 3D.

Query sets			Embedding in		
id			1D	2D	3D
1	TREC-5	Full	28.4	38.1 (+33.9*%)	39.9 (+40.4*%) (+4.9%)
2		Desc	23.6	44.2 (+87.8*%)	46.2 (+96.0*%) (+4.4%)
3		Title	22.0	36.8 (+67.3*%)	38.1 (+73.1*%) (+3.4%)
4		Title + Desc	22.2	35.3 (+59.2*%)	37.5 (+69.0*%) (+6.2*%)
5	TREC-6	Full	33.6	50.3 (+49.7*%)	51.4 (+53.1*%) (+2.2%)
6		Desc	30.7	51.4 (+67.4*%)	52.8 (+71.9*%) (+2.7%)
7		Title	29.6	51.1 (+72.6*%)	52.7 (+78.2*%) (+3.3%)
8		Title + Desc	32.7	50.0 (+52.8*%)	52.2 (+59.5*%) (+4.4%)
averaged on 2-8			27.8	45.6 (+64.2*%)	47.3 (+70.3*%) (+3.7*%)
averaged on 5-8			31.6	50.7 (+60.1*%)	52.3 (+65.2*%) (+3.2*%)

a significant improvement in quality when considering 2 and 3 dimensional visualizations (by 60.1% and 65.2% respectively). We also observe a statistically significant, though very small advantage of 3 dimensional visualizations over 2 dimensional ones (3.2%). This result confirms the findings made in the earlier study [16].

3.4 Search Strategy Effect

We experimented with several different search strategies that varied from a “simple-minded” Single Document strategy to significantly more sophisticated ones that adjusted their behavior as more relevant and non-relevant documents were discovered. We were interested in how much effect a search strategy has on the outcome of the simulation.

Both Table 3 and Table 4 report the embedding quality for two different strategies: the Cluster Centroid and the Single Document strategies. The former strategy makes intensive use of the relevant information and adapts as it discovers more relevant documents. The latter one is predetermined by the position of the starting point – the first relevant document. Therefore, it is not surprising that the former strategy tends to perform better than the latter one. What is surprising however, is that the difference is small: 1.3% for the 3 dimensional embeddings in Table 3.

Table 4 shows the precision numbers for a different initial condition. We assume that the relevance judgments for all top ten documents are known and the relevant documents among them are used as the starting points for the search strategies.

3.5 Comparison to Ranked List

Finally, the most important question that we considered in this study is how the quality of the visualization compares to the original ranked list. The earlier study [16] did not provide sufficient evidence that visualization has any advantages over the ranked list. Moreover, the authors of that study did not find any difference in performance between the visualization and the ranked list. Our experience with visualization suggests otherwise, so we were interested in seeing if we could improve on that result.

For this comparison we provide a second baseline: a ranking obtained by using relevance feedback [4]. Recall, that we assume the relevance judgments for at least one document – the highest ranked relevant – are known to the system. A search strategy elaborates on this information by using that document as a starting point from which it searches for the rest of the relevant material. We use the same relevance information to perform automatic relevance feedback, adjust the original query and *reorder* the existing ranked list. This approach is supposed to bring the relevant documents up in the ranked list and improving the user’s chance of finding them sooner.

Table 3 shows a significant advantage of the visualization over the ranked list (15.6% for 3 dimensional embeddings). The effect of relevance feedback to improve the ranked list is insignificant. Recall that we have

Table 3: Ranked list vs. spatial embeddings. Visualization quality evaluation of different query sets in different dimensions. The first column is for the system ranked list. The second column is for the system ranked list after relevance feedback was applied. The third column is for embeddings in 2- and 3-dimensions when precision is determined by the centroid search strategy. In the last column precision is determined by the single document search strategy. Percent improvement is from the ranked list to the embedding.

Query sets			Embedding					
					cluster centroid strategy		single document strategy	
id			RL	RF	2D	3D	2D	3D
1	TREC-5	Full	36.2	35.9 (-0.7 %)	38.1 (5.2 %)	39.9 (10.4*%)	37.4 (3.5 %)	39.3 (8.6 %)
2		Desc	34.1	42.6 (24.9*%)	44.2 (29.8*%)	46.2 (35.5*%)	42.0 (23.2*%)	44.0 (29.1*%)
3		Title	31.0	32.0 (3.3 %)	36.8 (18.9*%)	38.1 (23.0*%)	34.5 (11.5 %)	36.8 (18.8*%)
4		Title + Desc	31.0	34.5 (11.5*%)	35.3 (14.0*%)	37.5 (21.1*%)	34.1 (10.1 %)	36.9 (19.2*%)
5	TREC-6	Full	51.0	51.3 (0.6 %)	50.3 (-1.5 %)	51.4 (0.7 %)	50.3 (-1.4 %)	51.7 (1.4 %)
6		Desc	42.5	46.9 (10.3*%)	51.4 (20.9*%)	52.8 (24.2*%)	49.6 (16.8*%)	52.1 (22.6*%)
7		Title	40.1	42.4 (5.6 %)	51.1 (27.3*%)	52.7 (31.5*%)	48.4 (20.7*%)	51.4 (28.1*%)
8		Title + Desc	47.2	41.8 (-11.5*%)	50.0 (5.8 %)	52.2 (10.5*%)	48.4 (2.5 %)	51.2 (8.5 %)
averaged on 2-8			39.5	41.6 (5.3* %)	45.6 (15.2*%)	47.3 (19.5*%)	43.9 (11.0*%)	46.3 (17.1*%)
averaged on 5-8			45.2	45.6 (0.8 %)	50.7 (12.1*%)	52.3 (15.6*%)	49.2 (8.8* %)	51.6 (14.1*%)

only one relevant document for the relevance feedback. This is not condemnation of relevance feedback, but an illustration of its difficulty with few relevant documents.

If more relevant documents are available to a search strategy to use as starting points (Table 4) the advantage is more prominent (25.6% for 3 dimensional embeddings).

4 Conclusion and Future Work

In this paper we presented a substantial extension of the approach developed by Leuski and Allan [16]. We studied a visualization system for the purpose of helping the user to locate interesting material in the retrieved data. The system works by placing the documents into 1-, 2-, and 3-dimensional space and positioning them according to the inter-document similarity. As compared to the earlier study we provide a more robust threshold selection procedure that allows us to design a system that performs – on average – significantly better than traditional ranked list.

In addition to that we have observed that:

- Spatial structure of the embedding picture has a high correlation with the retrieval quality of the embedding. We defined a simple linear regression model that selects embeddings with significantly higher qualities than the ones uniformly selected.
- We confirmed earlier findings [16] that the dimensionality of the visualization plays an important role in defining its retrieval quality. We observed significant improvement of multidimensional visualizations over the 1 dimensional one. A 3 dimensional visualization has a small advantage over a 2 dimensional one.

4.1 Future Work

In this study we considered only two classes of documents: relevant and non-relevant. This was caused by the lack of data of any other kind. We are looking into extending our approach into situations when the user places the relevant documents into multiple classes. That task is modeled after the interactive TREC task of “aspect retrieval.”

Table 4: Ranked list vs. spatial embeddings. Visualization quality evaluation of different query sets in different dimensions. The top ten documents in the ranked list are judged. The first column is for the system ranked list. The second column is for embeddings in 2- and 3-dimensions when precision is determined by the centroid search strategy. In the last column precision is determined by the single document search strategy. Percent improvement is from the ranked list to the embedding.

Query sets			RL	Embedding			
id				cluster centroid strategy		single document strategy	
				2D	3D	2D	3D
1	TREC-5	Full	27.9	34.9 (24.8*%)	36.1 (29.0*%)	33.9 (21.2*%)	34.8 (24.6%)
2		Desc	25.6	36.7 (43.7*%)	37.9 (48.2*%)	34.8 (36.0*%)	36.1 (41.1%)
3		Title	25.6	30.7 (20.0*%)	32.9 (28.6*%)	28.6 (11.9*%)	30.2 (18.0%)
4		Title + Desc	19.6	28.1 (43.6*%)	29.8 (52.2*%)	27.1 (38.6*%)	28.5 (45.8*%)
5	TREC-6	Full	41.3	41.2 (-0.2%)	42.2 (2.2%)	40.6 (-1.7%)	41.2 (-0.4%)
6		Desc	34.6	46.7 (35.1*%)	49.0 (41.8*%)	44.7 (29.2*%)	47.1 (36.1*%)
7		Title	29.9	40.3 (34.5*%)	42.7 (42.6*%)	37.5 (25.2*%)	39.4 (31.5*%)
8		Title + Desc	35.4	40.8 (15.0*%)	43.5 (22.7*%)	39.0 (9.9%)	41.2 (16.2*%)
averaged on 2-8			30.3	37.8 (24.8*%)	39.7 (31.1*%)	36.0 (19.0*%)	37.7 (24.3*%)
averaged on 5-8			35.3	42.2 (19.6*%)	44.4 (25.6*%)	40.4 (14.5*%)	42.2 (19.5*%)

We are planning to do more work to investigate different user strategies before attempting a real user study. The user study is a useful final test of our hypotheses. We are also interested in visualizations that show how new documents relate to previously known material.

Acknowledgement

The earlier work referenced in this paper was done in collaboration with James Allan, who also took the time to review this paper. The author is grateful for his comments and suggestions.

This study is based on work supported in part by the National Science Foundation under grant number IRI-9619117. This material is based on work supported in part by the National Science Foundation, Library of Congress and Department of Commerce under cooperative agreement number EEC-9209623. This material is also based on work supported in part by United States Patent and Trademark Office and Defense Advanced Research Projects Agency/ITO under ARPA order number D468, issued by ESC/AXS contract number F19628-95-C-0235. Any opinions, findings and conclusions or recommendations expressed in this material are the author(s) and do not necessarily reflect those of the sponsors.

References

- [1] James Allan. *Automatic Hypertext Construction*. PhD thesis, Cornell University, January 1995. Also technical report TR95-1484.
- [2] James Allan. Building hypertext using information retrieval. *Information Processing and Management*, 33(2):145–159, 1997.
- [3] James Allan, Jamie Callan, Bruce Croft, Lisa Ballesteros, John Broglio, Jinxi Xu, and Hongmin Shu. Inquiry at TREC-5. In *Fifth Text REtrieval Conference (TREC-5)*, pages 119–132, 1997.
- [4] James Allan, Jamie Callan, W. Bruce Croft, Lisa Ballesteros, Donald Byrd, Russel Swan, and Jinxi Xu. Inquiry does battle with TREC-6. In *Sixth Text REtrieval Conference (TREC-6)*, 1998.
- [5] M. S. Barlett. The spectral analysis of two-dimensional point processes. *Biometrika*, 51:299–311, 1964.

- [6] Matthew Chalmers and Paul Chitson. Bead: Explorations in information visualization. In *Proceedings of ACM SIGIR*, pages 330–337, June 1992.
- [7] Noel A. C. Cressie. *Statistics for Spatial Data*. John Willey & Sons, 1993.
- [8] W. Bruce Croft. *Organising and Searching Large Files of Documents*. PhD thesis, University of Cambridge, October 1978.
- [9] David Dubin. Document analysis for visualization. In *Proceedings of ACM SIGIR*, pages 199–204, July 1995.
- [10] Thomas M. J. Fruchterman and Edward M. Reingold. Graph drawing by force-directed placement. *Software-Practice and Experience*, 21(11):1129–1164, 1991.
- [11] Donna Harman and Ellen Voorhees, editors. *The Fifth Text REtrieval Conference (TREC-5)*. NIST, 1997.
- [12] Donna Harman and Ellen Voorhees, editors. *The Sixth Text REtrieval Conference (TREC-6)*. NIST, 1998.
- [13] Marti A. Hearst and Jan O. Pedersen. Reexamining the cluster hypothesis: Scatter/gather on retrieval results. In *Proceedings of ACM SIGIR*, pages 76–84, August 1996.
- [14] M. Hemmje, C. Kunkel, and A. Willet. LyberWorld - a visualization user interface supporting fulltext retrieval. In *Proceedings of ACM SIGIR*, pages 254–259, July 1994.
- [15] Anton V. Leouski and W. Bruce Croft. An evaluation of techniques for clustering search results. Technical Report IR-76, Department of Computer Science, University of Massachusetts, Amherst, 1996.
- [16] Anton Leuski and James Allan. Interactive cluster visualization for information retrieval. In *Proceedings of ECDL'98*, September 1998.
- [17] Anton Leuski and James Allan. Strategy-based interactive cluster visualization for information retrieval. Technical Report IR-152, Department of Computer Science, University of Massachusetts, Amherst, 1998. Submitted for publication to *International Journal on Digital Libraries*.
- [18] G. Salton. *Automatic Text Processing*. Addison-Wesley, 1989.
- [19] C. J. van Rijsbergen. *Information Retrieval*. Butterworths, London, 1979. Second edition.
- [20] Jinxi Xu and W. Bruce Croft. Querying expansion using local and global document analysis. In *Proceedings of the 19th International Conference on Research and Development in Information Retrieval*, pages 4–11, 1996.