Scaling Spreading Activation for Information Retrieval

Anthony G. Francis, Jr., Mark Devaney, Juan Carlos Santamaria, Ashwin Ram
{centaur, markd, santmaria, ashwin}@enkia.com

Enkia Corporation
ATDC Suite N108
430 Tenth Street NW
Atlanta, Georgia 30318
http://www.enkia.com/

1. Introduction

The Information Retrieval Intelligent Assistant (IRIA) project applies principles of memory retrieval from cognitive science to the problem of information retrieval from large heterogeneous databases. IRIA uses spreading activation over a semantic network for information retrieval, a technique which has proven effective in a variety of tasks.

However, some of the very features which motivated the choice of spreading activation for information retrieval — such the use of fanout to automatically compute term weights, or the use of thresholds to automatically limit computation spent on irrelevant items — can introduce new problems as systems are scaled to larger sizes.

This paper discusses the use of semantic networks and spreading activation for information retrieval in the context of the IRIA approach, reviews some of the problems that arise as these technologies are scaled up to production systems, presents some preliminary results that illustrate these problems in practice, and discusses potential solutions.

2. The Problem

The success of the Net has made vast amounts of information on a wide variety of topics available all over the Earth. To fully exploit this rich but often frustrating resource, users require efficient and effective methods to retrieve information from what is essentially a large, heterogeneous, distributed database. A good solution to this problem would not only benefit users of the Web but also users of the vast intranets and datastores proprietary to large corporations and government organizations.

One of the primary challenges in this area of information retrieval is finding the wheat among the chaff — formulating queries which effectively focus retrieval on relevant information while excluding irrelevant information. Many techniques exist to tackle this problem, from precise query languages [2] to advanced relevance algorithms [3] to relevance feedback [13, 14]. The IRIA project combines the best features of many of these approaches.

2.1. The IRIA Approach

The Information Research Intelligent Assistant is an information retrieval architecture that addresses the problem of information retrieval from large heterogeneous databases. The IRIA approach is based on:

- dynamically harvesting semantic maps of information resources
- modeling user intent based on observed actions
- recommending useful information through implicit relevance feedback

An IRIA-based intelligent information management system acts as an autonomous assistant to a user working on a task, working unobtrusively in the background to learn both the user's interests and the resources available to satisfy those interests. This approach has proven effective at information retrieval in a variety of applications [5, 6].

2.2. Context-Sensitive Asynchronous Memory

The core of IRIA is based on a model called *context-sensitive asynchronous memory*, an approach to the problem of managing information access to large knowledge bases inspired by cognitive science research in human memory and expert performance [5].

Context-sensitive asynchronous memory is a semantic network / spreading activation approach similar in many ways to the declarative portion of the ACT architecture [1]. In a semantic network approach, knowledge is represented as a graph of nodes and links. Spreading activation uses this graph structure for memory retrieval: the source nodes in a query are given a certain amount of dynamic weight, or "activation", which is then iteratively propagated out along the links to other nodes in the network. Memory items are ultimately retrieved based on a function of their overall activation level.

Spreading activation has built-in limits in the form of fanout decay and propagation thresholds. The amount of activation propagated from a node decreases in inverse proportion to the node's fanout, or number of links, thus keeping the total amount of propagating activation constant. In some systems, activation stops propagating at some threshold, limiting the number of nodes that can be activated by a given source node [7].

The context-sensitive asynchronous memory approach builds upon this foundation, but is distinguished by the following features:

- a rich network representation in which each link between nodes instantiates some *relation* which is also a node
- a *context-directed spreading activation* process in which the activation of relation nodes alters the propagation of activation
- an *asynchronous retrieval monitor* which maintains a set of active retrieval requests which it constantly and incrementally attempts to satisfy

The first two features work together to enable the system to spend its search effort on parts of the knowledge base likely to be relevant; the second two features work together to enable the system to interleave memory search with other processes, permitting it to update its search with new information from reasoning.

Rather than computing activation directly from the influence of a set of activation sources, the context-directed spreading activation approach computes activation changes, or *perturbance*, from a list of *cues* maintained by the retrieval monitor. Perturbance propagates out into the network and cumulatively adds to activation, which in turn slowly decays. The list of cues itself is dynamic, changing over time as new information is added from reasoning. As a result, the activations in the network are influenced both by the history of cues seen and the current set of cues, enabling the system to shift the areas of the network it examines over time as the reasoning context changes.

The context-sensitive asynchronous memory architecture has been successfully applied to pure memory retrieval, to planning, to story understanding, and, in the IRIA system, to information retrieval.

2.3. The IRIA Architecture

IRIA builds upon the context-sensitive asynchronous memory approach by embedding it in a knowledge harvesting and interface monitoring architecture.

An IRIA system is a harvesting system: it exploits an existing data source, such as a search engine or database query interface, for the brute force work of indexing and searching the web. This metasearch approach enables IRIA to dynamically build a "semantic map"— a subset of the available information relevant to the user's interests.

An IRIA system is an interface monitor: it presents a view of the underlying data source as summarized by the semantic map, and monitors the actions users perform to inspect information in that map to develop a model of the user's intent. This intent model enables IRIA to dynamically track a user's interests in real time.

Together, these two faces of IRIA provide the necessary inputs to enable the context-sensitive asynchronous memory process to recommend useful information. The semantic map is represented directly within the rich semantic network representation of the context-sensitive asynchronous memory system, and the user intent model feeds cues to the asynchronous retrieval monitor. This enables an IRIA system to recommend information in the map to the user, or to draw new information in if the current map is insufficient.

The combined effect is a process of dynamic, implicit relevance feedback, in which user actions inform the system's knowledge of

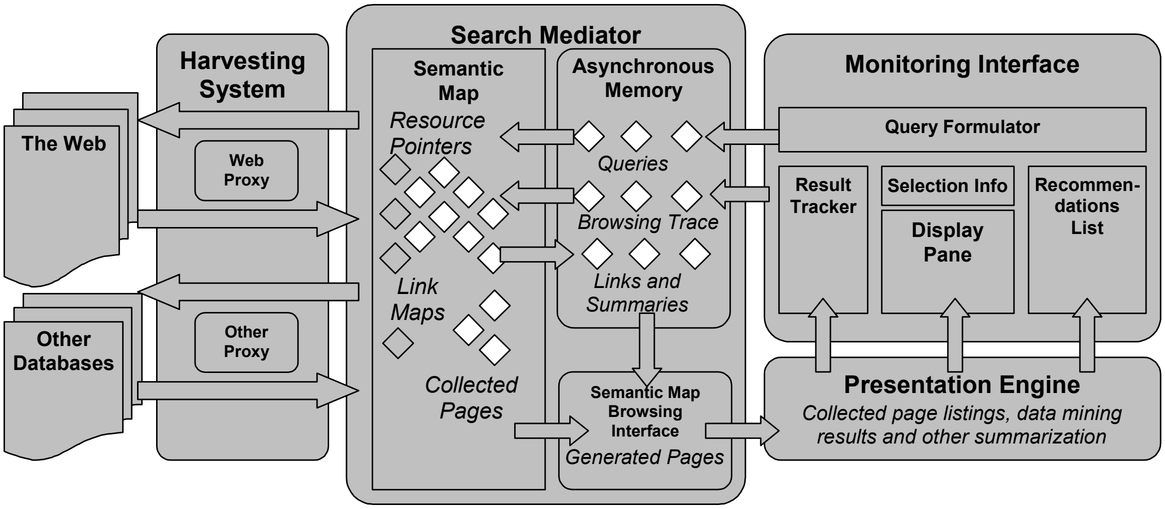


Figure 1. Architecture of IRIA

the user's interests and enable it to proactively recommend additional relevant information. The overall IRIA system architecture is shown in Figure 1.

2.4. Sources of Power

The effectiveness of the IRIA approach depends on both internal and external factors.

Data Input. The most important factor is external: quality data input. A typical IRIA system is deployed as an information research system augmenting a user's ability to navigate some large information resource, such as the Web or an intelligence database. Such a configuration will harvest data from some data source (such as a web search engine or database query mechanism) and observe actions at the user's interface (for example, a web search results form or an intelligence browser). The effectiveness of the harvesting algorithm used to construct the semantic map and the informativeness of the actions which can be observed to model intent will dominate the architecture's ability to deliver relevant information.

Algorithmic Factors. However, assuming that an IRIA system has a quality data source and can observe meaningful user actions, the quality of recommendations are then dominated by internal factors — the semantic map and spreading activation algorithms themselves.

These algorithms have a number of advantages. Their context-sensitive nature enables the network to be effectively partitioned for different queries. Unlike traditional systems which propagate spreading activation one-way from terms to documents and do not feed activation from one query into another, the CDSA approach actively exploits the patterns of activation which arise by allowing activation to reverberate through the network as a query's specification changes or when more than one query is active at one time.

While these algorithms have been shown to be an effective memory retrieval architecture for a variety of circumstances, information retrieval has peculiar properties which pose special challenges for the use of semantic networks and spreading activation for information retrieval.

3. Spreading Activation for IR

Spreading activation for information retrieval has a long history. While some interesting results have been demonstrated [4, 11], problems have been uncovered which limit the application of spreading activation to production-scale information retrieval systems. These issues arise in both the semantic networks which are the foundation for spreading activation and in the spreading activation process itself.

3.1. Properties of Information Retrieval

One of the most common forms of information retrieval is the problem of finding a document in response to a query. Documents and queries can take many forms, from short clippings to *War and Peace* and from one-word Web queries to complex logical specifications. Often, but not always, documents and queries can be effectively parsed apart into terms, and often, but not always, terms found in a document are evidence that the document is relevant to that query.

One of the most important features of information retrieval within this framework is that document collections can be very large. Well-known test collections contain tens of thousands or millions of documents spread out over gigabytes [2]. Natural collections are still larger: the Web contains over a billion documents [9] and some corporate intranets can be even larger.

Even with a harvesting and metasearch approach such as that used in the IRIA system, semantic networks and spreading activation can encounter problems with collections of this size. These issues affect both the construction of these networks and how they are searched.

3.2. Challenges for Semantic Networks

First and foremost, semantic networks are complex data structures with significant construction and storage requirements. Traditional inverted indices and vector-space representations can be directly constructed from document texts and connections between terms and documents are represented implicitly within parsimonious data structures. In contrast, semantic networks require some kind of ontology or content theory of how documents and terms are related. Whether this ontology is simple [12] or complex [4] it can make constructing the network a more complex proposition than a simple vector representation. Furthermore, a rich semantic network structure requires explicitly representing links between items in the network, which can be more space expensive than a more restricted representation.

3.3. Challenges for Spreading Activation.

Spreading activation in turn has its own problems for information retrieval. Salton examined a simple semantic network for information retrieval, in which documents and terms are nodes linked when a document contains a term, and showed that it captures only part of the information found in the typical *tfidf* term weighting used in many vector-space information retrieval systems [12]. As a result, the traditional computation of spreading activation with fanout at each node [7] automatically computes inverse document frequency, but does not directly represent term frequency, and as a result this naïve formulation does not perform as well as the traditional vector space representation [12]. Of course, a richer network structure or different link weights could address some of these concerns.

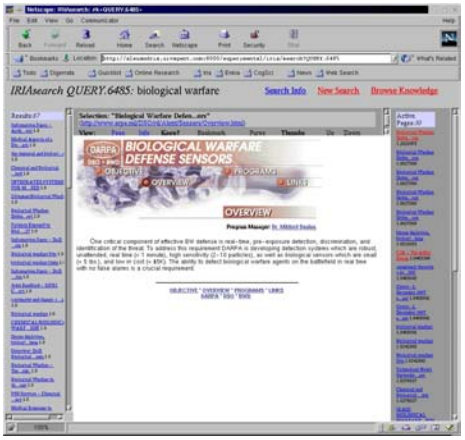
3.4. Challenges for Context-Directed Spreading Activation.

Context-directed spreading activation, the core of the IRIA approach, attempts to address the scalability limits of traditional spreading activation by contextually guided flow of activation. Activation thresholds are key to context guidance, enabling the system to not spend effort perturbing the activations of irrelevant portions of the knowledge base. When thresholding is applied to the information retrieval domain it causes ‘automatic stopwording’ of high-frequency terms, such as ‘the’ and ‘and’ which appear in almost every document.

Unfortunately, as the semantic map grows in size, the fanout of useful words will grow as well. This limits the amount of information which can be drawn into the semantic map in response to a single query on any given topic. This furthermore limits the map as it accumulates over time: if a user has a persistent interest in a single topic, words about that topic will increase in frequency and will become stopworded. Ultimately, this can make a system effectively ‘blind’ to topics that the user is interested in — precisely opposite the intent of the approach.

3.5. Speed-Accuracy Tradeoffs

Essentially, this limit of spreading activation is a speed-accuracy tradeoff. Fanout and thresholds are employed in spreading activation to place an upper bound on the memory retrieval computation performed for any one piece of evidence (any one cue). In both traditional and context-directed spreading activation, this threshold acts to prevent spread of activation when the amount of activation spread is very small and will have little effect on the relative weights of items in the knowledge base and hence little effect on retrieval. In other words, spreading activation with fanout thresholds reduces retrieval cost at the price of a slight


File Edit View Go Communicator Help
Back Forward Reload Home Search Netscape Print Security Stop
Bookmarks Location What's Related
Search Info
New Search
Browse Knowledge
View: Page Info Keep? Bookmark Purge Thumbs Up Down
DARPA
OBJECTIVE
PROGRAMS
OVERVIEW
LINKS
100%

IRIAsearch QUERY.6485: biological warfare

Selection: "Biological Warfare Defen...ors"

BIOLOGICAL WARFARE DEFENSE SENSORS

OVERVIEW

Program Manager

One critical component of effective BW defense is real-time, pre-exposure detection, discrimination, and identification of the threat. To address this requirement DARPA is developing detection systems which are robust, unattended, real time (< 1 minute), high sensitivity (2-10 particles), as well as biological sensors which are small (< 5 lbs.), and low in cost (< $5K). The ability to detect biological warfare agents on the battlefield in real time with no false alarms is a crucial requirement.

Figure 2. Prototype IRIA Interface

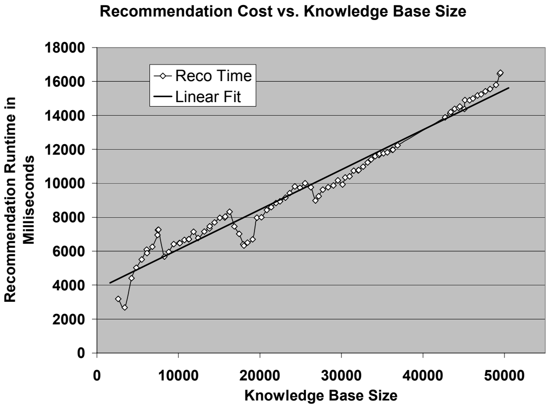


Figure 3. Prototype Scaleup of Recommendation Cost

reduction in accuracy.

Therefore, spreading activation with fanout thresholds becomes potentially less accurate for information retrieval tasks as document collections of increasing size, but this accuracy tradeoff itself is scalable: the threshold cutoff can be changed at the price of increased computational effort.

4. Dealing with Scalability

The IRIA project has applied a variety of techniques to deal with these scalability issues. These techniques include developing representations and algorithms which make it feasible to use semantic networks and spreading activation for information retrieval, as well as techniques to address the specific limitations of spreading activation and context-directed spreading activation on larger document sizes.

4.1. Representations and Algorithms

The algorithms and representations used in the IRIA project have developed over time. The original IRIA prototype was developed in Lisp, deployed as an extension to the CL-HTTP web server [10], and accessed through a browser-based interface. The prototype uses a metasearch system to execute a query on existing Web search engines (e.g., Altavista, Yahoo, etc.) and then summarizes the returned hits into a knowledge map. The prototype application displays search results on the left and the user's current selected result in the center. As the user browses, IRIA is reminded of pages and displays these dynamically computed results on the right, enabling users to quickly focus on relevant results (Figure 2).

Prototype Performance. This prototype had a number of flaws. Its semantic map consumed a large amount of memory, requiring megabytes worth of storage per thousand nodes. Its raw speed and performance were poor, taking several seconds to harvest information and several seconds to perform recommendations. Furthermore, the cost of recommendations increased with knowledge base size. While this scaleup had a low constant factor it still was a linear complexity in knowledge base size for an algorithm which was designed for constant-time complexity in knowledge base size. Figure 3 shows a typical recommendation time curve for the IRIA Lisp prototype, charting the number of milliseconds necessary to compute a recommendation cycle against the total number of nodes in the knowledge base.

Production Version. Since the prototype, the IRIA system has been reimplemented. The new edition of IRIA, marketed under the name "Enkion", is still accessed through a browser-based interface similar to Figure 2 but is now implemented in Java and deployed on an Enterprise Java Beans application server.

The production version of IRIA had a variety of features which addressed the flaws of the prototype. The semantic map employed a new representation which eschewed explicit representation of nodes as Java objects as much as possible in favor of arrays of connections, resulting in a hundredfold decrease in size. The spreading activation algorithms were also

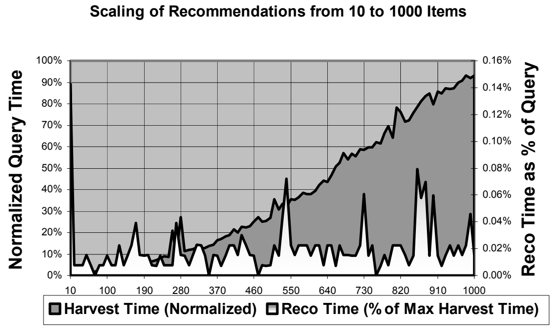


Figure 4. Production Scaleup of Recommendation Cost

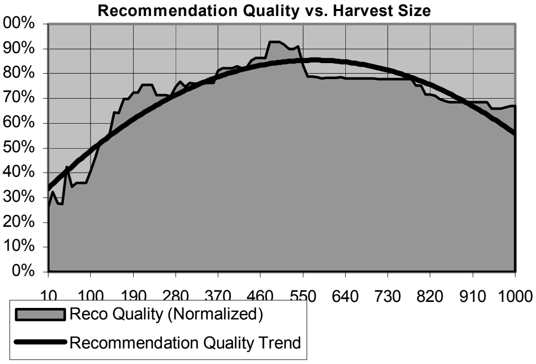


Figure 5. Production Scaleup of Recommendation Quality

updated, resulting in a tenfold to a hundredfold increase in recommendation speed.

4.2. Assessing Scalability

To address the question of scaleup of recommendation cost, we conducted a preliminary experiment testing how recommendation cost and quality were affected as the amount of information harvested into a map increased.

Method. The method for this experiment consisted of simulating a query-recommendation episode in a web search application. In this simulated episode, the system was presented with a query and was then allowed to harvest information related to that query into the semantic map from a web search engine (AltaVista). The system was then presented with topical information to seed the user intent model and then was allowed to generate recommendations. This simulates a user who enters an underspecified query (such as “centaur”) but who is really interested in more specific information (such as the Centaur upper-stage rocket launch vehicle).

The dependent variables in this experiment were the choice of query terms, the number of search results harvested and the choice of topic. For simplicity, the initial assay chose one query term (“centaur”), and chose three topics relevant to the query term (“rocket”, “asteroid”, “mythology”). The harvest size was allowed to vary from 10 to 1000 items.

The independent variables measured were time to harvest results, time to recommend results, and quality of recommendations. Harvest result times and recommendation times were measured in milliseconds through lightweight profiling code embedded into the recommendation core. The system was tested on the Sun Java 1.2.2 platform hosted on a 500Mhz Pentium III with 256Mb RAM running Windows NT.

For simplicity in the initial assay, and to reduce the possibility of human error, recommendation quality was measured comparatively against a traditional vector space model. In this test, the quality of a recommendation was measured by its vector space similarity to the terms in the topic stored in the user intent model, and the aggregate quality of recommendations was computed by averaging this score over the top ten recommendations of produced by IRIA to the topic terms provided as part of the user intent model.

Obviously, this method is limited — if the vector space and context-directed spreading activation model “disagree”, this metric will penalize the CDSA model even if a human might judge the CDSA recommendations to be of high quality. However, this eliminated the possibility that human judges’ subjective views could distort the results, and provided an inexpensive “ballpark” metric that could be rapidly computed over a wide range of harvest sizes.

The hypothesis tested in these experiments were that harvest time would increase linearly with harvest size, recommendation time would remain fixed with harvest size, and recommendation quality would initially increase with harvest size, but might experience a dropoff as fanout of key words increased

Results. Speed results were as predicted. Harvest time increased approximately linearly from 0.02 seconds for 10 items to 0.93 seconds for 1000 items (averaged across all runs). Recommendation time was difficult to measure because the time taken by each recommendation cycle was often less than the granularity of the profiling code (10ms); however, no significant increase was detected as the knowledge base size increased. Figure 4 plots harvest speed normalized against the longest harvest time in dark grey and presents recommendation time as a percentage of the longest harvest time in white.

Quality results were also as predicted. While each individual topic had a different effect on recommendation quality, averaged across all three topics there was a definite increase in quality of recommendations up to about 500 harvested items. Beyond 500 items, a drop in quality was observed (at least as measured in terms of the vector space similiarity of recommendations to the target topic). Figure 5 illustrates this phenomenon.

Discussion. Analysis of this behavior is still ongoing, but by examining the properties of the IRIA algorithms, we can make educated guesses about the conditions under which a quality dropoff will be observed.

The configuration of the IRIA system used in these experiments imposes a fanout threshold which enables each cue to activate approximately 1000 nodes (initial activation 1.0, threshold 0.001). As the number of documents in the harvest increases, words which appear in many documents will be contribute less and less to the ordering of the recommendations.

Furthermore, recommendations in an IRIA system depend on activation reverberating through a network, which as a consequence must involve activation which has been diluted by fanout. As the fanout of key nodes at later stages increases, they too may become thresholded, causing entire sections of the network to effectively "drop" from their original positions in the recommendations.

5. Conclusion

Further analysis of the scaling properties of the production IRIA system are underway, using additional queries and topics and additional data sets with larger harvest values. While this work is ongoing, we can point to ways to address the limitations this research has uncovered in applying spreading activation to information retrieval. This could include revised parameterizations which lower the propagation threshold or increase the initial activation of certain terms, or revised network structures which take high-fanout words related to user interests and break them apart into lower-fanout concepts using structures such as redundant discrimination nets [8].

Acknowledgements

This work was conducted at Enkia Corporation with the support of an Air Force Rome Labs SBIR Phase II contract. Additional support was provided by the Department of Education, Cox Interactive Media and the Georgia Institute of Technology.

References

[1] Anderson, J. R. (1983a). *The Architecture of Cognition.* Cambridge, Massachusetts: Harvard University Press.

[2] Baeza-Yates, R. & Ribeiro-Neto, B. (1999). *Modern Information Retrieval*. Addison-Wesley-Longman.

[3] Brin, L. & Page, S. (2000). The Anatomy of a Large-Scale Hypertextual Web Search Engine. In *Proceedings of the 7th International World Wide Web Conference*, Brisbane, Australia,

[4] Cohen, P.R. & Kjeldsen, R. (1987). Information retrieval by constrained spreading activation in semantic networks. *Information Processing and Management*, 23, pp.255-268.

[5] Francis, A. (2000). Context-sensitive asynchronous memory. Ph.D. Dissertation, College of Computing, Georgia Institute of Technology.

[6] Francis, A.G., Ram, A., and Devaney, M. (2000). IRIA: The Information Research Intelligent Assistant. In Arabina, H.R., (Ed.) *Proceedings of the International Conference on Artificial Intelligence IC-AI'2000*, Las Vegas, Nevada, June 26-29, 2000. CSREA Press.

[7] Jones, R. M. (1989). *A model of retrieval in problem solving.* Doctoral dissertation. Department of Information and Computer Science, University of California, Irvine.

[8] Kolodner, J.L. (1984). *Retrieval and organization strategies in conceptual memory: a computer model*. Northvale, NJ: Lawrence Earlbaum Associates.

[9] Lawrence, S. & Giles, C.L. (1999). Accessibility of information on the web.

[10] Mallery, J.C. 1994. A Common LISP hypermedia server. Proceedings of the First International Conference on the World-Wide Web, Geneva: CERN, May 25, 1994.

[11] Pirolli, P. & Card, S. K. (1999). Information foraging. *Psychological Review* October 1999, (106) 4.

[12] Salton, G. & Buckley, C. (1988). On the Use of Spreading Activation Methods in Automatic Information Retrieval. Technical Report, *TR88-907*, April 1998, Cornell University.

[13] Salton, G. & Buckley, C. (1990). Improving retrieval performance by relevance feedback. *Journal of the American Society for Information Science*, 41, pp.288-297.

[14] Sparck Jones, K. (1979) Search term relevance weighting given little relevance information. Journal of Documentation, 35, 30-48. Reprinted in Sparck Jones, K. & Willet, P. (Eds.) (1997). *Readings in Information Retrieval.* San Francisco: Morgan Kaufmann.