Application-Embedded Retrieval from Distributed Free-Text Collections

Vladimir A. Kulyukin
School of Computer Science, Telecommunications,
and Information Systems
DePaul University
http://kulyukin.cs.depaul.edu

Abstract

A framework is presented for application-embedded information retrieval from distributed free-text collections. An application's usage is sampled by an embedded information retrieval system. Samples are converted into queries to distributed collections. Retrieval is adjusted through sample size and structure, anydata indexing, and dual space feedback. The framework is investigated with a retrieval system embedded in a text processor.

The Problem

The integration of query generation and feedback challenges information retrieval (IR) technologies. Relevance feedback (Aalbersberg 1992; Robertson and Walker 1997), while useful to information science professionals (Spink and Saracevic 1997), is frequently inappropriate for lay users. Many users are unable to adequately state their information needs in queries (Burke, Hammond, and Young 1996). The constant, explicit solicitation of relevance judgments interferes with many users' information-seeking behaviors (Kulyukin 1998c). Feedback utilization in the query space alone prohibits retrieval adjustments over multiple interactions (Kulyukin 1998a; Kulyukin, Hammond, and Burke 1998). The focus on maximizing the number of relevant documents in every retrieval is inconsistent with many users' information needs (Jansen et al. 1998).

The approach to the integration of query generation and feedback henceforth presented is called *application-embedded distributed IR*. The approach is implemented in an embedded IR (EMIR) system for text processors. The objective is a technology for building non-intrusive, feedback-sensitive IR tools that users embed into their applications to tap into and monitor information sources, while engaged in routine usages of those applications. The technology is motivated by a growing number of sources with a wealth of data, but few tools to timely put it to use.

A Solution

An application-embedded IR system, such as EMIR, is deployed through *resource subscription* and *mode selection*. During resource subscription, the user specifies distributed resources from which information is retrieved. During mode selection, the user specifies how the application is used. For example, EMIR has a LaTeX mode in which the text processor is used for typesetting LaTeX documents. As the user typesets a document, EMIR takes background text samples and converts them into queries to the subscribed collections. The retrievals are examined by the user at the user's convenience. The following techniques form the system's core:

- **Background Sampling**: Two types of sampling are distinguished: *random* and *structured*. In random sampling, the entire sample is chosen randomly. In structured sampling, samples are components of known templates, e.g., abstracts and references of LaTeX papers. Structured sampling is based on the assumption that some document components carry more information than others (Bradshaw, 1998; Kulyukin, Hammond, and Burke 1996). Query generation from background sampling does not disrupt the user.
- **Anydata Indexing**: The vector space retrieval model is used (Salton and McGill 1983). A document is a vector of its terms' weights. A document collection is a vector space whose dimensions are the documents' terms. In smaller collections, a term's rarity is its main distinguishing property. In larger collections, the term's pattern of occurrences in the documents prevails. Since indexing is so adjusted to the currently available data, it is called *anydata*.
- **Dual Space Feedback**: Document space relevance feedback modifies term weights in the document vectors from relevance judgments (Brauen 1971). Over long periods, some terms acquire negative weights, which adversely affect performance (Robertson and Walker 1997). Since only the dimensions' weights are modified, query terms that are not dimensions have no impact (Kulyukin 1998c). Dual space feedback addresses both problems with two vector spaces, *pos-*

itive and *negative*, both of which represent the same document collection. When a document is relevant to a query, the weights are adjusted in the positive space (p-space). When a document is nonrelevant, the same adjustments are made in the negative space (n-space).

- **Distributed Indexing**: The Common Object Request Broker Architecture (http://www.omg.org) (CORBA) connects the system to distributed resources. Interobject communication occurs through Object Request Brokers (ORB's) via the Internet Inter-ORB Protocol (IIOP). A resource is a server offering a set of indexing and retrieval services. Collection indexing and maintenance are delegated to the servers.

Details

Terms

The terms are computed from queries, i.e., background free-text samples of LaTeX papers, and the documents in each collection. The Xerox Part-of-Speech tagger (ftp://parcftp.xerox.com/pub/tagger/) assigns each term stochastically determined parts of speech (stpos) (Cutting et al. 1992). The tagged text is stoplisted with an extended version of the Brown Corpus stoplist (Francis and Kucera 1982). Terms are normalized through a greedy morphological analysis (Kulyukin 1998c). A list of rules for the term's tag is searched. Each rule has the form `{match, add, id}`, where `match` and `add` are strings, and `id` is an integer. A rule applies when `match` and the term's suffix match, whereupon the suffix is removed, and `add` is added. When no rules apply, the term is left as is. The normalization procedure is implemented on top of the morphological kernel of WordNet^{TM} (Miller 1995) (ftp://ftp.cogsci.princeton.edu/pub/wordnet/).

Term weights

A term's weight combines its semantic and statistical weights. The semantic weight is intrinsic. The statistical weight depends on the collection's size and the term's distribution in the documents. The semantic weight of a term t_i, $\omega_{sm}(i)$, is inverse to its polysemy and proportional to its stpos: $\omega_{sm}(i) = \pi(i)^{-1}\rho(i)$, where $\pi(i)$ is t_i's polysemy in WordNet, and $\rho(i)$ is its stpos weight. Nouns are valued at 1.0, verbs and adjectives at .5, and adverbs at .25.

The statistical weight unifies two approaches: *inverse document frequency* (IDF) (Salton and McGill 1983) and *condensation clustering* (CC) (Bookstein, Klein, and Raita 1998; Kulyukin, Hammond, and Burke 1998). IDF values a term's rarity in the collection; CC values terms' nonrandom distribution patterns over the collection's documents. Let D be the total number of documents. Define $f(i,j)$ to be t_i's frequency in the j-th document d_j. Put $\tilde{n}_j = 1$ if $f(i,j) > 0$, and 0, otherwise. Put $D_i = \sum_{i=1}^{D} \tilde{n}_i$. For t_i's IDF-weight, put $\omega_{idf}(i) = A_{idf} + log(D/D_i)$, with A_{idf} a constant. For t_i's $tfidf$ weight in d_j, put $\omega_{tfidf}(i,j) = f(i,j)(A_{idf} + log(D/D_i))$. The CC-weight of t_i is the ratio of the actual number of documents containing at least one occurrence of t_i over the expected number of such documents: $\omega_{cc}(i) = A_{cc} + log(E(\tilde{D}_i)/D_i)$, where A_{cc} is a constant and $\tilde{D}_i$ is a random variable assuming D_i's values. Put $T_i = \sum_{j=1}^{D} f(i,j)$. Since $\tilde{n}_i$ assumes 1 and 0 with the respective probabilities of p_i and $q_i = 1 - p_i$, $E(\tilde{n}_i) = p_i = 1 - (1 - 1/D)^{T_i}$. Since $\tilde{D}_i = \sum_{i=1}^{D} \tilde{n}_i$, $E(\tilde{D}_i) = Dp_i$. For t_i's $tfcc$ weight in d_j, put $\omega_{tfcc}(i,j) = f(i,j)\omega_{cc}(i)$. Let $A_{cc} = A_{idf}$. By definition, $\omega_{cc}(i) = \omega_{idf}(i) + logp_i$. Hence, the lemma: If $A_{cc} = A_{idf}$, $\omega_{cc}(i) = \omega_{idf}(i) + \log p_i$. The lemma brings a closure to the discussion of the relationship between IDF and CC (Bookstein, Klein, and Raita 1998; Kulyukin, Hammond, and Burke 1998). A class of metrics obtains, unifying IDF and CC: $\omega_{idfcc}(i) = A + B\omega_{idf}(i) + Clogp_i$, where A, B, and C are constants. If $A = A_{idf}$, $B = 1$, and $C = 0$, $\omega_{idfcc} = \omega_{idf}$; if $A = A_{cc}$, $B = 1$, and $C = 1$, $\omega_{idfcc} = \omega_{cc}$. Since $f(i,j)$ approximates the importance of t_i in d_j, t_i's weight in d_j is given by $\omega_{tfidfcc}(i,j) = f(i,j)\omega_{idfcc}(i)$. Combining t_i's semantic and statistical weights gives t_i's weight in d_j: $\omega_{cw}(i,j) = \omega_{sm}(i)^{\alpha_1}\omega_{tfidfcc}(i,j)^{\alpha_2}$, where α's reflect the component weights' importance.

Feedback

A collection is represented by two vectors: the p-space centroid (p-centroid) and the n-space centroid (n-centroid). Centroid weights are the average weights of the space's dimensions. Two vector spaces are stored locally: one of p-centroids and one of n-centroids. A query is turned into a term vector (q-vector). The similarity is the cosine of the angle between the q-vector and a centroid. The similarity between the q-vector and a p-centroid is *positive*. The similarity between the q-vector and an n-centroid is *negative*. A collection is relevant if the difference between its positive and negative similarities exceeds a threshold. When the collection is relevant, the query is marshalled to its server. The retrievals are merge-sorted locally by the similarity to the query. Relevance judgments cause adjustments in the appropriate space. When a document is relevant, the adjustments are made in the p-space; otherwise, they are made in the n-space. The adjustment procedure is the same for both spaces: the similarities are rewarded, the differences are punished, and the new terms are added to the vectors to encourage or discourage future retrievals. This approach to feedback utilization is similar to negative evidence acquisition (Kulyukin, Hammond, and Burke 1998), a technique for adjusting content representations in response to feedback. However, it stays within one model, precludes negative weights, and avoids the proliferation of ad hoc representations.

Evaluation

The experimental data were taken from the collection of requests for comments (rfc's) maintained

slen vs. qsize	10	20	30
$avslen_1$	10.1	7.5	6.4
$avslen_2$	12.4	8.9	7.7
$avslen_3$	15.7	11.3	8.3

slen vs. part	abst			intro			refs		
$avslen_1$	8.5	5.1	3.2	8.1	4.7	3.4	14.4	11.1	8.7
$avslen_2$	8.9	7.3	3.7	9.2	7.5	7.1	15.7	14.5	12.3
$avslen_3$	11.1	8.1	4.7	11.9	7.9	8.5	18.2	18.1	15.4

Figure 1: Random vs. structured queries with dual space feedback

by the IETF, the Internet Engineering Task Force (ftp://ftp.isi.edu/in-notes/). 1423 rfc's were manually partitioned into three topics: data transport protocols, applications protocols, and data content. Each topic constituted a collection. Each collection was managed by a CORBA server on a separate CPU. EMIR was implemented as a CORBA client. Each server published its services through the Interface Definition Language (IDL) (http://www.omg.org). The services included indexing a collection, retrieving top n matches under a given similarity metric, retrieving all matches above a threshold, computing centroids, adjusting term weights, adding new terms, retrieving rfc texts, and several other operations. The inputs were 20 papers randomly selected from a pool of 100 LaTeX papers on network protocols. In each experiment, the following mode was used. A subject with a background in network protocols was asked to typeset each paper as if it were his own. The subject would, at his convenience, inspect the retrieval folder and volunteer relevance judgments on retrieved rfc's. Two options, relevant and nonrelevant, were available. To keep the feedback solicitation nonintrusive, the subject was not obligated to render judgment on every retrieved rfc. To preclude the fatigue factor, each paper was typeset on a different day.

Metric

The evaluation metric used in all experiments was the average search length, $avslen_n$, i.e., the number of nonrelevant rfc's before the n-th relevant one. For example, if $n = 1$, the numbers of nonrelevant rfc's before the 1st relevant one in all retrieved sets are added and divided by the number of submitted queries. The two standard evaluation metrics, *precision* and *recall* (Harman 1996; Salton and McGill 1983), were not used. Recall was not used, because its assumption that *all* relevant documents are known for *every* query is not realistic for large, dynamic collections. Such relevance counts, while useful in theory, are not feasible in practice. A growing body of experimental evidence (Burke et al. 1997; Jansen et al. 1998; Kulyukin 1998c) suggests that most users (80-90%) examine only the top 2-4% of retrievals. Consequently, little benefit is gained from retrieving all relevant documents if none of them makes it to the top. Precision was not used, because, while it does consider relevant documents with respect to the actual retrievals, it fails to consider *where* in the retrieved set the relevant documents are. A retrieved set of 20 documents with the relevant documents occurring in the last two positions has the same precision as a set of 20 documents with the relevant documents occurring in the first two positions. In EMIR, empty retrieved sets are probable. If precision is used, such sets must be treated as special cases inasmuch as precision is undefined if nothing is retrieved.

Results

The first experiment tested the differences between random and structured sampling. In random sampling, queries were computed from random term samples of sizes 10, 20, and 30. The system did the greedy morphological analysis of the paper's text typeset to this point, ignoring the LaTeX commands, and randomly selected the required number of terms. In structured sampling, queries were computed from samples taken from specific parts of the paper: abstract, introduction, and references. To compute a sample of size n from a part P, the system parsed the text for P, compute the total number of terms (T) in it, chose a random position i, $0 \leq i \leq T - n - 1$, take n consecutive terms beginning at i, and applied the greedy morphological analysis to the selection. The sizes for structured samples were the same as for random samples. When sampling from references, the terms were taken from titles. The top 20 rfc's were retrieved in response to every query. Three metrics $avslen_1, avslen_2$, and $avslen_3$ were computed. A retrieved set with at least one relevant rfc had a minimum search length of 0 and a maximum search length of 19. Sets with no relevant rfc's had the search length of 20. For each of the 20 test papers, 20 queries (10 random and 10 structured) were computed for each query size (qsize). The term weight metric was $\omega_{cw}(i,j) = \omega_{sm}(i)^{\alpha_1}\omega_{tfidfcc}(i,j)^{\alpha_2}$, where $\alpha_1 = \alpha_2 = 1$. The experiment's results are presented in Figure 1. The top table gives the random queries' avslen's. The bottom table gives the structured queries' avslen's for the three document parts. Thus, the values in the $avslen_1$ row under *abst*, i.e., 8.5, 5.1, and 3.2, are the $avslen_1$'s for the qsizes 10, 20, and 30, respectively, computed from the abstracts. The top table sug-

slen vs. qsize	10	20	30
$avslen_1$	11.4	8.5	7.0
$avslen_2$	12.7	10.7	7.9
$avslen_3$	17.7	15.4	10.3

slen vs. part	abst			intro			refs		
$avslen_1$	10.1	6.2	4.2	8.9	5.2	4.2	14.7	11.3	8.7
$avslen_2$	10.7	7.9	5.8	10.1	8.4	7.8	15.9	14.7	12.5
$avslen_3$	14.3	9.4	7.7	13.0	10.7	9.3	18.6	18.4	15.7

Figure 2: Random vs. structured queries without dual space feedback

gests that the avslen improves as the random queries' sizes increase. Thus, random queries may be a viable backup when no structure is available because of the nature of the data or parsing failures. The bottom table suggests that the introduction of structure into sampling makes a difference. The avslen's for the queries computed from abstracts and introductions are lower than their random counterparts. No significant difference was found between the abstract and introduction queries. A closer analysis revealed that many papers had an overlap between the terms in the abstract and introduction. Ideas were stated in short sentences in the abstract and repeated or expanded with the same terms in the introduction. The reference queries performed worse than their structured and random counterparts. One explanation is that the titles may not provide sufficient descriptions of the papers' content (Bradshaw, 1998). Another explanation is that authors may cite each other for political reasons, e.g., to promote a research agenda, for which EMIR cannot account.

The second experiment tested the dual space feedback. The queries and relevance judgments from the first experiments were taken, and the avslen's were recomputed with the dual space feedback disabled. As the tables in Figure 2 show, feedback disablement had a negative impact on avslen's. The change was pronounced with abstract and introduction queries. It was less noticeable with random and reference queries. A closer analysis indicated that the dual space feedback prevented nonrelevant rfc's from moving ahead of the relevant ones.

The third experiment tested anydata indexing. The experiment's objective was to find evidence to support the hypothesis that in larger collections, both homogeneous and heterogeneous, a term's tendency to condense in documents is a better predictor of relevance than its rarity. Two document collections, homogeneous and heterogeneous, were used. The homogeneous collection included 432 rfc's on TCP, UDP, and IP. The heterogeneous collection included the homogeneous collection plus 510 rfc's on Mail and Usenet message formats. The similarity metric, queries, and relevance judgments were the same as in the first two experiments. The performance metric was $avslen_3$. For each collection, the values of A and C ran between -30 and 30 in steps of .25; B was set to 1. Fixing B at 1.0 and varying A and C was aimed at testing the effects of the constant and the correction on IDF. If the best performance were to be found at $C = 0$, it would indicate that the tendency to condense is not as good a content predictor as rarity. In the first collection, the best $avslen_3$ was 9.21, when $A = -1$ and $C = -1.2$; the best value of $avslen_3$ for IDF, i.e., $A = B = 1$, and $C = 0$, was 12.74. In the second collection, the best $avslen_3$ was 8.2 when $A = -1.25$ and $C = -20$; the best value of $avslen_3$ for IDF was 15.25. Thus, in the larger heterogeneous collection, CC performed better than IDF. In both collections, the correction's effect was greater than the effect of the constant. The effect was more pronounced in the larger heterogeneous collection. One explanation is that a larger collection is likely to cover different topics. If terms are valued by their tendency to condense in the documents pertinent to their topics, content-bearing terms have a better chance to deviate from being randomly distributed over the documents.

Discussion

The integration of query generation and feedback continues to attract the attention of AI and IR researchers (Burke, Hammond, and Young 1996; Fitzgerald 1995; Kulyukin, Hammond, and Burke 1998). The presented approach shares with this research the intuition that better integrations are achieved in task-embedded systems. However, application-embedded IR differs from its counterparts in several respects.

Since indexing is distributed, there is no central content repository that is either static (Burke, Hammond, and Young 1996; Fitzgerald 1995), or must be periodically maintained with expensive reindexing operations (Burke et al. 1997; Kulyukin 1998b; Kulyukin 1998c). Information sources maintain themselves and offer an IR system a set of services agreed upon in advance. To index a source is not to build and maintain an adequate representation of its content, but to know what services the source offers and when to use them.

No explicit query generation is required of the user. Since many lay users find it hard to put their information needs into queries, let alone into formalisms of modern IR systems, queries are automatically generated by the system from sampling the application's us-

age. In this respect, EMIR is similar to example-based knowledge navigation systems, such as FindMe systems (Burke, Hammond, and Young 1996), which monitor how users navigate in information spaces. However, EMIR differs from FindMe systems, because query generation occurs in the background, while in FindMe systems it depends on the user explicitly critiquing multiple examples before the right one is found.

Feedback solicitation is nonintrusive. Such feedback solicitation is consistent with the empirical evidence that, unlike information science professionals, lay users almost never give feedback when it is solicited explicitly (Jansen et al. 1998; Spink and Saracevic 1997). Feedback utilization is persistent. Unlike many other IR systems that use relevance feedback in the query space alone (Aalbersberg 1992; Harman 1996), EMIR uses feedback to modify its representations in the document space. The user is assumed to be consistent in his or her relevance judgments over a long period of time. The assumption is valid, because EMIR targets one user, whereas similar systems target large and heterogeneous user groups that are unlikely to be consistent in their judgments.

Conclusion

A framework for application-embedded retrieval from distributed free-text collections was developed and illustrated with EMIR, an IR system embedded in text processor. Queries are generated from background samples of free-text papers and used in retrievals from distributed collections. Feedback is never solicited explicitly, and is utilized persistently only when volunteered. A family of term weight metrics was presented, unifying IDF and CC. It was shown how retrieval can be adjusted through background sampling, anydata indexing, and dual space feedback. Each technique was evaluated and found useful.

References

Aalbersberg, I. J. 1992. Incremental Relevance Feedback. In Proceedings of the 15th Annual International SIGIR Conference, 11-21.

Bookstein, A.; Klein, S. T.; and Raita, T. 1998. Clumping Properties of Content-Bearing Words. *Journal of the American Society for Information Science* 49(2):102-114.

Bradshaw, S. G., 1998. Reference Directed Indexing: Attention to Descriptions People Use for Information. Masters thesis, Dept. of Computer Science, The University of Chicago.

Brauen, T. L. 1971. Document Vector Modification. In G. Salton, editor, *The SMART Retrieval System: Experiments in Automatic Document Processing*, 456-484. Englewood Cliffs, NJ: Prentice-Hall, Inc.

Burke, R. D.; Hammond, K. J.; Kulyukin, V.; Lytinen, S. L.; Tomuro, N.; and Schoenberg, S. 1997. Question Answering from Frequently Asked Question Files: Experiences with the FAQ Finder System. *AI Magazine* 18(2):57-66.

Burke, R. D.; Hammond, K. J.; and Young, B. C. 1996. Knowledge-based Navigation of Complex Information Spaces. In Proceedings of AAAI-96.

Cutting, D.; Kupiec, J.; Pederson, J.; and Sibun, P. 1992. A Practical Part-of-Speech Tagger. In Proceedings of the Third Conference on Applied Natural Language Processing. Trentano, Italy: Association of Computational Linguistics.

Fitzgerald, W. 1995. Building Embedded Conceptual Parsers. Ph.D. diss., Dept. of Computer Science, Northwestern University.

Francis, W., and Kucera, H. 1982. *Frequency Analysis of English Usage.* New York: Houghton Mufflin.

Harman, D. 1996. Overview of the Fourth Text REtrieval Conference. In Proceedings of the Fourth Text REtrieval Conference (TREC-4).

Jansen, B. J.; Spink, A.; Bateman, J.; and Saracevic, T. 1998. Real Life Information Retrieval: A Study of User Queries on the Web. *SIGIR Forum* 32(1):5-18.

Kulyukin, V. 1998a. FAQ Finder: A Gateway to Newsgroups' Expertise. In Proceedings of the 40th Conference of Lisp Users.

Kulyukin, V. 1998b. An Interactive and Collaborative Approach to Answering Questions for an Organization. In Proceedings of the ASIS-98 Mid Year Conference.

Kulyukin, V. 1998c. Question-Driven Information Retrieval Systems. Ph.D. diss., Dept. of Computer Science, The University of Chicago.

Kulyukin, V.; Hammond, K.; and Burke, R. 1998. Answering Questions for an Organization Online. In Proceedings of AAAI-98.

Kulyukin, V.; Hammond, K.; and Burke, R. 1996. Automated Analysis of Structured Online Documents. In Proceedings of the AAAI-96 Workshop on Internet-Based Information Systems.

Miller, G. A. 1995. WordNet: A Lexical Database for English. *Communications of the ACM* 38(11):39-41.

Robertson, S., and Walker, S. 1997. On Relevance Weights with Little Relevance Information. In Proceedings of ACM SIGIR.

Salton, G., and McGill, M. 1983. *Introduction to Modern Information Retrieval.* New York: McGraw-Hill.

Spink, A., and Saracevic, T. 1997. Interactive Information Retrieval: Sources and Effectiveness of Search

Terms During Mediated Online Searching. *Journal of the American Society for Information Science* 48(8):741-761.