A Vector Space Equalization Scheme for a Concept-based Collaborative Information Retrieval System

Takashi Yukawa
Nagaoka University of Technology
1603-1 Kamitomioka-cho, Nagaoka-shi
Niigata, 940-2188 JAPAN

Sen Yoshida and **Kazuhiro Kuwabara**
NTT Communication Science Laboratories
Nippon Telegraph and Telephone Corpotation
2-4 Hikaridai, Seika-cho
Kyoto, 619-0237 JAPAN

Abstract

This paper describes a vector space equalization scheme for a concept-based collaborative information retrieval system; evaluation results are given. The authors previously proposed a peer-to-peer information exchange system that aims at smooth knowledge and information management to activate organizations and communities. One problem with the system arises when information is retrieved from another's personal repository since the framework's retrieval criteria are strongly personalized. The system is assumed to employ a vector space model and a concept-base as its information retrieval mechanism. The vector space of one system is very different from that of another system, so retrieval results would not reflect the requester's intention. This paper presents a vector space equalization scheme, the automated relevance feedback scheme, that compensates the differences in the vector spaces of the personal repositories. A system that implements the scheme is realized and evaluated using documents on the Internet. This paper presents implementation details, the evaluation procedure, and evaluation results.

Introduction

The authors previously proposed a peer-to-peer information exchange system (Yukawa, Yoshida, & Kuwabara 2002). The aims of the system include smooth knowledge and information management to activate organizations and communities. The conventional server-centric systems are weak because they create information-provisioning bottlenecks. Given this background, the proposed framework targets the collaborative interworking of personal repositories that accumulate per-user information, and accept service requests. The framework will lead to peer-to-peer-type (Oram 2001) information sharing systems that exchange documents stored in the personal computers of users. In this paper the systems are called peer-to-peer collaborative personal repository systems. The framework can potentially resolve the information-providing bottlenecks and accomplish smooth information sharing for organizations and communities.

Locating information in documents stored in personal repositories is a prerequisite of information sharing and exchange. Thus, information retrieval (IR) plays a very important role in any personal repository system. Current systems employ a vector space model with a concept-base as their IR mechanism, because such models can be tightly personalized to the user. However, a problem arises when collaborative IR is established in distributed and isolated environments. In a collaborative system, each individual stores documents according to his/her own expertise, interests, and likes (we call these "world views") in his/her own personal repository. Although these documents are written in a natural language, the vocabulary differs from user to user. This means that the vector space of one system is quite different from that on another system, which renders the retrieval result from another user's repository inadequate. For instance, let us assume that user A is looking for information on portable information appliances. Let us also assume that he/she considers "notebook computer" and "PDA (Personal Digital Assistant)" to be the same thing, that is, his/her repository contains many documents in which these two words co-occur. In the concept base generated from the repository, these two words lie close to each other in user A's vector space. In addition, let us assume that user B, who is not interested in PDAs, has a concept base that maps "notebook computer" and "desktop computer" close to each other in user B's vector space. When user A wants to search for information on portable information appliances from his/her own personal repository, he/she simply inputs "notebook computer" as the query keyword to retrieve the desired documents. On the other hand, if user A wants to search user B's personal repository and inputs "notebook computer" into user B's system, a lot of the results include documents that do not discuss PDAs. Instead, many results address the desktop PC, and so are not desirable.

This means that we need schemes for collaborative IR that can assimilate world views to improve the effectiveness of searches across different personal repositories. We have already proposed a framework for vector space equalization schemes that will use collaborative IR to solve the problem (Yukawa, Yoshida, & Kuwabara 2002).

This paper proposes a detailed procedure for one such scheme, the automated relevance feedback scheme, and describes its implementation. Experimental results gained by using real press release documents posted on the Internet are also reported. The results confirm that the scheme will yield systems that provide more effective retrieval results.

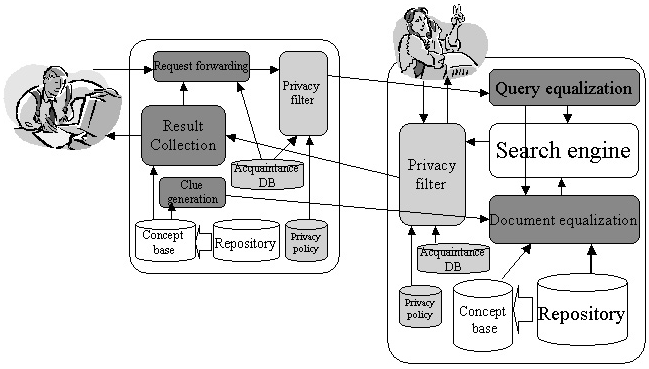

Figure 1: Peer-to-Peer Collaborative Personal Repository System

Background

Collaborative Personal Repository System

A system for storing, retrieving, and manipulating privately collected information is called a personal repository. Conventional information storage and retrieval systems target the handling of large volumes of centralized information, a good example is a library. Obviously, the personal repository differs from the conventional systems in terms of contents and intention. It focuses on information acquired personally and intended for personal exploitation. It is more like a person's bookshelf than a library. Haystack (Adar, Karger, & Stein 1999) is a widely known research and development project targeting personal repositories.

The most significant feature of personal repository systems, indeed a fundamental difference from library-like systems, is adaptation to the owner to support the retrieval and/or manipulation of information stored locally. To achieve this, a semi-structured database capable of storing the information plus its attendant metadata is employed as the storage module. Each system also has an IR mechanism that can be personalized to reflect the owner's interests. Haystack does not have a built-in search engine and instead assumes the use of off-the-shelf tools. We believe that the concept-based IR system described below is suitable as a search engine for a personal repository.

We have proposed a framework for the peer-to-peer collaborative personal repository system shown in Figure 1. A stand alone system that implements the framework has, as common components, a repository to store information collected by the owner, a concept base generated with the method described in the next subsection, and a search engine that exploits the concept base. As described in the previous section, the system needs to contain an IR mechanism to interact with other personal repositories. The components of the collaborative IR mechanism are represented by the shaded boxes in the figure.

IR with Concept-based Vector Space Model

As described above, the IR for a personal repository has to reflect the owner's world views. We earlier proposed and developed an IR system comprising an extended vector space for an expert recommendation system (Yukawa *et al.* 2001). We believe that it can also be used as a search engine for a personal repository because its concept-base, which is used to form the vector space, can be adapted to handle the bulk of the information stored in the system. This subsection introduces the system briefly.

Expressing documents and queries as vectors in a multi-dimensional space and taking the relevance or similarity as the cosine coefficient between two centroid vectors is known as the Vector Space Model (Salton & Buckley 1998). In a basic relevance discernment scheme that exploits the vector space model, the vector of a document can be mapped onto a hyper-space where each keyword in the set of documents corresponds to an axis; the values along the axes for the documents correspond to the TF$\times$IDF values of the keywords in the documents. Because the scheme assumes a vector space in which the keywords directly correspond to the axes, there is, however, the problem that synonyms and/or co-occurrences of keywords are not considered.

Some improved methods that can solve the above problem have been proposed. One such method, the concept base, is grounded in co-occurrence Schütze (Schütze & Pedersen 1994; 1995). This method first counts the word co-occurrences in close proximity in the documents and then constructs a word co-occurrence matrix. Second, it reduces the rank of the matrix using Singular Value Decomposition (SVD) to yield the keywords' vector space. We call it here the "Concept-base." The vector for a document is represented as the sum of the keyword vectors generated from it. In this method, documents having similar content have strong relevance even if the documents do not use the same expressions. This differs from methods based on word occurrences, or a boolean full-text search, in which high relevance is obtained only when the documents use similar expressions. It should be pointed out that concept-based relevance discerning methods allow keywords and documents, which are fundamentally different from each other, to be mapped together in the same multi-dimensional space. This means that the methods provide not only relevance between keywords, but also relevance between keywords and documents, and between documents.

An IR system is basically a system that calculates and stores document vectors for every target document in advance and offers a list ordered in terms of relevance between the query keywords and the target documents. As queries, such systems will accept not only keywords but also a list of keywords or even documents. Therefore, it can retrieve documents for a query keyword, documents for a query document, or keywords for a query document. We call such a system an "IR System with Concept-based Vector Space Model."

IR with Vector Space Equalization

Vector Space Equalization

As described above, the vector spaces of personal repositories differ from each other. Because of this, applying own query keywords to another personal repository is not assured of returning results that will satisfy the user. Figure 2 illus-

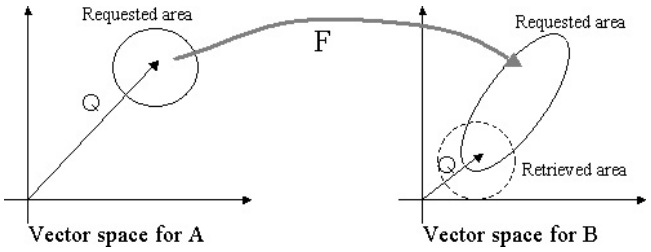

Figure 2: Vector Space Mapping

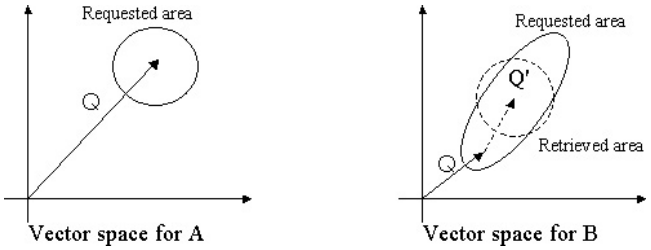

Figure 3: Automated Relevance Feedback Method

trates this problem. The search engine looks for documents inside a hyper-sphere having as its center the query vector in vector space. The hyper-sphere in the vector space of user A is projected to the hyper-ellipse in the vector space of user B. The query vector of user A is also mapped in the figure. Because the search engine of user B's system looks for documents inside the hyper-sphere having as its center the query vector in B's own vector space, the area of results obtained by the system differs from user A's desired area, i.e., the area of A's hyper-ellipse. Therefore, vector space equalization is used to transform the search space in user B's system so that better results can be obtained. A vector space equalization scheme is presented in the following subsection.

Automated Relevance Feedback Scheme

As illustrated in Figure 3, positioning a query vector at the center of a hyper-ellipse raises the probability of obtaining desirable documents even though the retrieved area is still shaped like an ellipse. This idea, query vector equalization, is derived from the technique known as relevance feedback. Conventional relevance feedback is a human-to-computer feedback technique in which a human judges whether documents retrieved by a computer are desirable or not. On the other hand, the proposed scheme judges the adequacy of the documents by using the concept base to establish the feedback loop automatically.

The IR procedure in the proposed scheme is described below, where P_A denotes the system owned by the requester, i.e., user A, and P_B denotes the system owned by the user providing the information, i.e., user B.

1. P_A transfers its query keywords to P_B when user A submits a request.
2. P_B retrieves documents based on its own concept base. Assume that the result includes documents $D_1, D_2, \ldots, D_n$ where n is the number of documents in the result and document vector d_i corresponds to document D_i. Next, P_B returns $D_1, D_2, \ldots, D_n$ to P_A.
3. P_A evaluates the received results based on its own concept base. That is, the relevance between document D_i and the query is computed by assessing its own concept-base. Assuming that the relevance degree of document D_i is RA_i,
4. P_A labels each document in the result acceptable or not depending on its relevance degree. We define J_i as

$$J_i = \begin{cases} \alpha & (\text{if } RA_i > T) \\ \beta & (\text{otherwise}) \end{cases} \quad (1)$$

where T is an appropriate threshold value, and α and β are feedback coefficients.

5. P_A sends $J_1, J_2, \ldots J_n$ to P_B.
6. P_B adjusts the query vector according to the relevance feedback technique. That is, the query vector in P_B is modified based on the judgment. Denoting the current query vector as $\vec{Q_B}$ and the new query vector as $\vec{Q_B'}$, $\vec{Q_B'}$ is expressed as:

$$\vec{Q_B'} = \vec{Q_B} + \sum_{i=1}^{n} J_i d_i. \quad (2)$$

7. P_B retrieves documents using the new query vector, and returns the results to P_A.
8. Steps 3 and 7 are iterated until the appropriate conditions are satisfied.

Because P_A notifies P_B of the adequacy of the documents, the vector space in P_A is never estimated by P_B. On the other hand, P_A can learn a lot of information about P_B's repository because P_B provides P_A with not only relevant documents but also other documents for the purpose of judging adequacy. Therefore, in this scheme, P_A can accomplish its searches without disclosing its world views, whereas P_B has to disclose some portions of its world views.

Implementation

Figure 4 illustrates a structure of a personal agent.

The white boxes are the common components of a stand alone IR system that uses the concept-based vector space model. Morphological analysis is used to divide the target documents into a set of keywords. The co-occurrence values of the keywords are determined and a word co-occurrence matrix $\mathbf{D}$ is generated. Assuming that the number of keywords is M, the size of $\mathbf{D}$ is $M \times M$. Next, SVD is used to reduce the number of dimensions of $\mathbf{D}$ to obtain an $M \times L$ matrix $\mathbf{C}$. This matrix is the concept-base, in which the i-th row corresponds to the vector for the i-th keyword, $\vec{k_i}$. Empirical examinations indicate that the optimal dimensionality of the keyword vector in the concept base, L, lies in the range of 100 to 200 (Yukawa *et al.* 2001).

Using the concept-base generated above, a document vector for each document in the personal repository is calculated

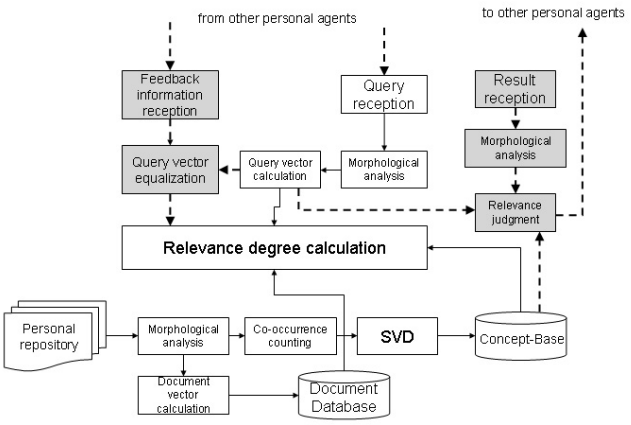

Figure 4: Structure of a Personal Agent

and stored in the document database together with the document itself. The document vector, $\vec{D_x}$, is defined as the sum of the keyword vectors. That is:

$$\vec{D_x} = \sum_{i=1}^{N_x} \vec{k}_{K(w_{xi})}, \tag{3}$$

where $(w_{x1}, w_{x2}, ..., w_{xN_x})$ denotes the keywords of document D_x, and $K(w)$ denotes a function that obtains the index number corresponding to keyword w.

Given a retrieval request from the owner, the system uses morphological analysis to divide the query into a list of keywords and calculates the query vector $\vec{Q}$ as sum of the keyword vectors. For every document in the document database, the relevance degree values of the query vectors and the document vectors, R_{qx}, are taken as the cosine coefficients of the vectors, that is:

$$R_{Qx} = \frac{\vec{Q} \cdot \vec{D_x}}{|\vec{Q}| \times |\vec{D_x}|}. \tag{4}$$

The documents are sorted in descending order of the relevance degree, and the top N documents are returned as the result, where N is given by the user.

If the retrieval request was sent from another personal agent, automated feedback components, the shaded boxes in Fig. 4, are activated. Assuming P_A to be a requester's personal agent and P_B to be a personal agent of the target, P_B uses morphological analysis to divide the query from P_A into a list of keywords and then calculates its query vector. Relevance degrees for the query and each document are calculated and documents included in the retrieval result are selected using procedures created for stand alone retrieval. The retrieval result is returned to P_A.

After receiving the result, P_A calculates the relevance degrees between the query and every document in the result by referring to its own concept-base. Because the concept-bases for P_A and P_B are different, the relevance degree between the query and a document in P_A differs from that in P_B. P_A generates a list of the IDs of documents that have greater relevance degree than a given threshold T_h, and sends it to P_B. P_B recalculates the query vector according to the feedback procedure described in the previous section and performs the retrieval procedure again. This feedback procedure is iterated as needed.

Evaluation and Considerations

To validate the improvement in retrieval precision possible with this scheme, we performed an experiment using a document set consisting of press releases and review articles on PCs and PDAs. This section describes the experimental procedure and the results.

Experimental Procedure

The target document set consisted of manufacturer's press releases and review articles related to computer equipment and posted on the Internet. The documents were categorized into two groups. One consisted of documents on mobile computing appliances (including light-weight notebook PCs, PDAs and mobile phones), while the other consisted of documents on all sorts of PCs. We defined the former as document set D_A, and the latter as document set D_B. Special document set D_S was also defined. It consisted of documents on a specific model of PDA.

We created personal repository P_A containing document sets D_A and D_S, and P_B holding D_B and D_S. That is, the owner of P_A is very interested in notebook computers and PDAs, while the owner of P_B is interested in all sorts of PCs. The vectors of keywords related to notebook computers and PDAs are supposed to be located closely in the concept-base of P_A, while those related to notebook computers and desktop computers are supposed to be located closely in the concept base of P_B. D_A D_B each had 1000 documents while D_S had 500 documents. Therefore, there were 1500 documents in each personal repository.

Using these document sets, we conducted the following experiment.

1. P_A retrieves N_A documents for the query "notebook PC." Let the result be R_A; the set of documents included in the result that belong to D_S is S_A.
2. P_B retrieves N_B documents for query "notebook PC." Let the result be R_{B0}.
3. P_A calculates the relevance degree values between the query and each document in R_{B0} and returns the IDs of the documents that have greater relevance than the given threshold T_h.
4. P_B follows the proposed scheme to compensate its query vector and retrieves the documents again. Let the result be R_{B1}.
5. Recall and precision for $R_{Bi}(i = 0, 1)$ are defined as

$$r_i = \frac{|S_A \cap R_{Bi}|}{|S_A|}, \tag{5}$$

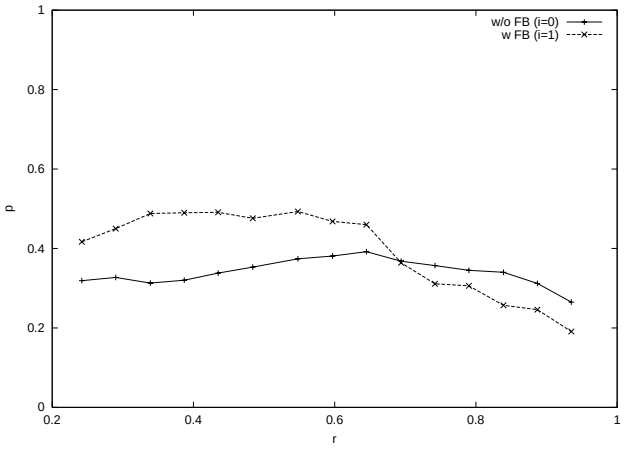

Figure 5: Experimental Results

$$p_i \quad = \quad \frac{|S_A \cap R_{Bi}|}{|R_{Bi}|}. \qquad (6)$$

6. Examine (r, p) pairs for various N_B.

Results and Considerations

Figure 5 shows the recall–precision plot of the above experiment, where $N_A = 100, T_h = 0.2, \alpha = 1.0, \beta = 0.6$. The solid line indicates the result without vector space equalization, while the dashed line plots that with the proposed scheme. As demonstrated in the figure, the proposed scheme achieves a precision of around 0.5 when $r \leq 0.6$, while the corresponding value is 0.3 without vector space equalization. This means that only $1/3$ of the retrieved documents are desirable if IR is performed without vector space equalization; the proposed scheme doubles the ratio of desired document to 0.5.

As r exceeds 0.7, the precision possible with vector space equalization falls. The recall of $r = 0.7$ is obtained when the number of retrieved documents $N_B = 120$. Similarly, $r = 0.8$ and $r = 0.9$ are obtained when $N_B = 150$ and $N_B = 200$ respectively. This means that documents that have rather low relevance degree must be included in the result to obtain the recall rates noted. Those low-ranked documents basically have little content on the query topic, "notebook PC", and yield relevance degree values equivalent to irrelevant documents. Thus, the ranks of those documents are very sensitive to subtle differences in the structure of the vector space. This fact suggests that the precision rate is unstable if the recall is large. Therefore, this phenomenon is not taken to reflect an intrinsic weakness of the vector equalization scheme.

Summary and Future Works

This paper studied collaborative IR for a peer-to-peer personal repository system targeting the smooth sharing of personally collected information. It proposed and evaluated a highly effective vector space equalization scheme. Because IR criteria on the system are tightly personalized, simply transferring query requests does not provide desirable results. To solve this problem, we developed an IR function that searches for documents in equalized vector spaces.

A scheme for vector space equalization, the automated relevance feedback scheme, was detailed and evaluated. For evaluation, we implemented an IR system, and collected document sets from press releases and review articles related to computer equipment. The experimental results confirm that the proposed scheme improves the recall rate of the retrieval results.

Because the experiment was intended to simply validate the feasibility of the vector equalization scheme, only one type of documents and one query were used. The feedback coefficients were fixed to values that were assumed to be appropriate. A future task is conduct more detailed evaluations with various suites of documents and queries.

It is important to note that the scheme proposed in this paper performs vector space equalized retrieval through the modification of the query vector instead of manipulating the vector space directly. One of our prior papers introduced a vector space equalization scheme that equalizes the structure of the vector space by space mapping estimation (Yukawa, Yoshida, & Kuwabara 2002). Confirming the performance of that scheme experimentally and comparing it to that of the scheme proposed in this paper are other goals.

References

Adar, E.; Karger, D.; and Stein, L. 1999. Haystack: Per-user information environment. In *Proc. 1999 Conference on Information and Knowledge Management*, 413–422.

Oram, A. 2001. *Peer-to-Peer*. O'Reilly.

Salton, G., and Buckley, C. 1998. Term weighting approaches in automatic text retrieval. In Jones, K. S., and Willet, P., eds., *Readings in Information Retrieval*. Morgan Kaufmann Publishers. 323–328.

Schütze, H., and Pedersen, J. O. 1994. A cooccurrence-based thesaurus and two applications to information retrieval. In *Proc. RIAO '94*.

Schütze, H., and Pedersen, J. O. 1995. Information retrieval based on word sense. In *Proc. 4th Annual Symposium on Document Analysis and Information Retrieval*, 161–176.

Yukawa, T.; Kasahara, K.; Kato, T.; and Kita, T. 2001. An expert recommendation system using concept-based relevance discernment. In *Proc. International Conference on Tools with Artificial Intelligence 2001*, 257–264.

Yukawa, T.; Yoshida, S.; and Kuwabara, K. 2002. Collaborative information retrieval for a personal repository system based on vector space model. In *Worknote of Pacific Rim International Workshop on Multi-Agents*, 101–112.