Applied Artificial Intelligence, 16:283–302, 2002
Copyright © 2002 Taylor & Francis
0883-9514/02 $12.00 + .00

□ MINING A WEB CITATION DATABASE FOR DOCUMENT CLUSTERING

Y. HE, S. C. HUI, and A. C. M. FONG
School of Computer Engineering, Nanyang Technological University, Singapore

The World Wide Web has become an important medium for disseminating scientific publications. Many publications are now made available over the Web. However, existing search engines are ineffective in searching these publications, as they do not index Web publications that normally appear in PDF (Portable Document Format) or PostScript formats. One way to index Web publications is through citation indices, which contain the references that the publications cite. Web citation Database is a data warehouse to store the citation indices. In this paper, we propose a mining process to extract document cluster knowledge from the Web Citation Database to support the retrieval of Web publications. The mining techniques used for document cluster generation are based on Kohonen's Self-Organizing Map (KSOM) and Fuzzy Adaptive Resonance Theory (Fuzzy ART). The proposed techniques have been incorporated into a citation-based retrieval system known as PubSearch for Web scientific publications.

Many scientific publications are now available online over the Internet or stored in the form of digital libraries (Schatz and Chen 1996). However, they tend to be poorly organized, making the search of relevant research publications difficult and time consuming. Commercial search engines such as Yahoo!, Lycos, and Excite have been developed to help users to locate information of their interest by matching queries against the database of stored indexed documents. However, these search engines are ineffective for searching scientific publications accurately. This is due to the fact that most research documents, which are mainly in PostScript or PDF (Portable Document Format) formats, are not normally indexed by the commercial search engines.

In this research, we have developed a citation-based retrieval system known as PubSearch (He 2000), which generates a Web Citation Database from online scientific publications that are made available over the Internet,

Address correspondence to Y. He, School of Computer Engineering, Nanyang Technological University, Nanyang Avenue, Singapore 639798.

and supports retrieval of the publications based on the Web Citation Database. The Web Citation Database is a data warehouse used for storing citation indices, which contain the references that the publications cite. Such citations are used to refer the reader to the relevant papers for further reading on the concepts and ideas that are introduced in the source paper. These cited references reveal how the source paper is linked to prior relevant research on the assumption that citing and cited references have a strong link through semantics. They provide a valuable source of information and directives for researchers in the exchange of ideas, the current trends and future development in their respective fields. Therefore, citation indices can be used to facilitate searching and retrieval of relevant research information.

The Web Citation Database can be generated using an autonomous citation indexing agent by searching through Web sites on the Internet. The agent downloads the scientific publications, extracts the citations, generates citation indices and stores the information in the Web Citation Database. This technique has also been demonstrated in another system known as CiteSeer (Bollacker, Lawrence, and Giles 1998, 2000). CiteSeer can convert PostScript and PDF documents to text using *pstotext* from the Digital Virtual Paper project (DEC 2000). The citation indices created are similar to the Science Citation Index (Garfield 1979). As such, the Web Citation Database can be generated to contain citation information for Web publications. When new publications are made available on the Web, the indexing agent will be able to find them and store them into the Web Citation Database.

The Web Citation Database contains rich information that can be mined for the retrieval of scientific publications over the Internet. In this paper, we focus on mining (Fayyad, Piatetsky-Shapiro, and Smythe 1996; Mitchell 1999) the Web Citation Database for document clustering to group related papers into clusters. Two kinds of clustering techniques, namely the Kohonen's self-Organizing Map (KSOM) (Kohonen 1995) and Fuzzy Adaptive Resonance Theory (Fuzzy ART) (Carpenter, Grossberg, and Rosen 1991), are investigated. The clustering results are then incorporated into the PubSearch system for the retrieval of Web publications.

DOCUMENT CLUSTERING TECHNIQUES

Clustering algorithms can be broadly divided into two basic categories: hierarchical and non-hierarchical algorithms (Kaufman and Rousseeuw 1990). Hierarchical clustering algorithms involve a tree-like construction process. Among various hierarchical clustering algorithms, Agglomerative Hierarchical Clustering (AHC) algorithm (Jain, Murty, and Flynn 1999) is probably the most commonly used. This algorithm is sensitive to halting criteria as the stopping point greatly affects the results. Too early or too late combination of clusters often causes poor results. Non-hierarchical clustering algorithms select the cluster seeds first and assign objects into clusters based on the seeds specified. The seeds may be adjusted accordingly until all clusters are stabilized. These algorithms are faster than AHC algorithms. The K-means algorithm (Rocchio 1966) is a non-hierarchical clustering algorithm that can produce overlapping clusters. However, its disadvantage is that the selection of initial seeds can have a great impact on the final result. Several variants of the K-means algorithm have been reported in the literature. One of them is the Leader Clustering algorithm (Hartigan 1975), which selects the initial partition by assigning the first data item to a cluster and considers the next data item by measuring the distance between the new item and the existing cluster centroids. This process repeats until all data items are clustered.

Recently, many other document clustering algorithms have been proposed, including Suffix Tree Clustering (Zamir and Etzioni 1998), Supervised Clustering (Aggarwal et al. 1999), and Word Clustering (Slonim and Tishby 2000). Suffix Tree Clustering is a linear time clustering algorithm based on identifying the phrases that are common to groups of documents as opposed to other algorithms that treat a document as a set of unordered words. In contrast to all other clustering algorithms, which are unsupervised clustering, Supervised Clustering assumes that a pre-existing sample of training documents with the associated classes is available in order to provide the supervision of the categorization of the whole document collection. A set of seeds, which are representative of those classes, are identified and served as the starting points of the subsequent clustering process. The subsequent clustering process is independent of any further supervision. The Word Clustering method is quite different from all other clustering algorithms as it incorporates the information bottleneck method (Tishby, Pereira, and Bialek 1999). It comprises two stages. First, word clusters are extracted based on the distribution of the documents in which they occur. In the second stage, the original representation of the documents is replaced by a much more compact representation based on the co-occurrence of word clusters in the documents. Using this new document representation, the same clustering procedure for word clusters is then re-applied to obtain the desired document clusters.

In addition, various models of the artificial neural networks have been applied to document clustering. Some of the well-known examples include Kohonen's Self-Organizing Map (KSOM) (Kohonen 1995) and Adaptive Resonance Theory (ART) (Carpenter, Grossberg, and Rosen 1991) models. Competitive learning is required in these neural networks. However, the learning or weight update procedures are quite similar to those in some classical clustering approaches. For example, KSOM is essentially a stochastic version of K-means clustering method (Jain, Murty, and Flynn 1999). The only difference between KSOM and K-means is that, besides the closest cluster, the neighboring clusters are updated as well in KSOM. The learning algorithm in ART models is similar to the Leader Clustering algorithm (Jain, Murty, and Flynn 1999).

The KSOM algorithm has been applied to information retrieval (Lin 1997; Honkela et al. 1998; Rauber and Merkl 1999). It can be used to display a colorful map of topic concentrations, which can be further explored by drilling in to browse the specific topic. This is demonstrated in the WEBSOM (Honkela et al. 1998; Kohonen et al. 2000) system. Maps are provided to give a visual overview of the whole document collection, with similar documents located close to each other. The computational complexity of constructing the mapping function in KSOM is $O(M^2)$, where M denotes the number of model vectors (Kaski et al. 1998). However, in the special case where the ratio between the "width" of the neighbourhood and the size of the map is fixed, the computational complexity is only O(M). So, there is a trade-off between the computation time and the size of the map M that determines the resolution of the mapping.

However, KSOM is not very suitable for the Web publication environment. The Web Citation Database is dynamic as new scientific publications can be created anytime. Whenever a new publication is added into the document collection, the learning process needs to be performed on the new document collection again, making this algorithm computationally expensive. In this regard, Fuzzy ART should be a better choice as it allows continuous learning and does not require re-learning for the whole document collection. However, the Fuzzy ART networks are order-dependent, that is, different clusters are obtained for different orders in which the data are presented to the network. Also, the size and the number of clusters generated by Fuzzy ART depend on the value chosen for the vigilance threshold, which is used to decide whether a pattern is to be assigned to one of the existing clusters or to start a new cluster. In this paper, both the KSOM and Fuzzy ART algorithms are investigated as mining techniques for the Web Citation Database. Different from WEBSOM, the KSOM algorithm is applied to the Web Citation Database instead of Usenet newsgroups or patent abstracts to generate document clusters.

CITATION-BASED RETRIEVAL

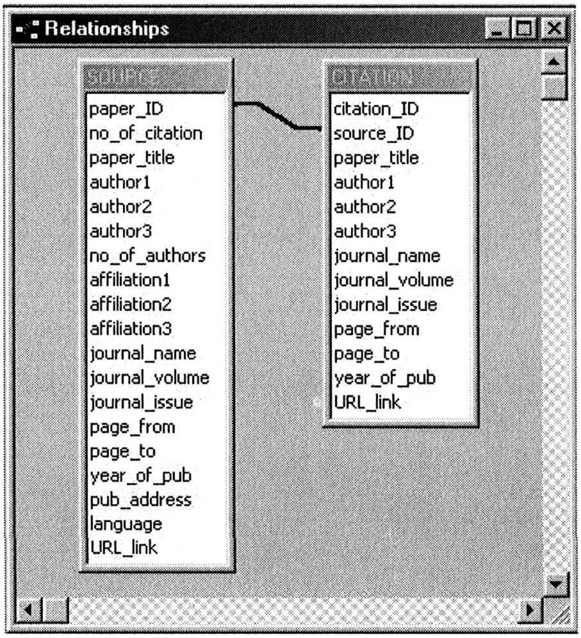
To our knowledge, there are only two systems that support citation-based document retrieval, one is provided by the Institute for Scientific Information (ISI) (ISI 2000), and the other is CiteSeer (Bollacker, Lawrence, and Giles 1998; Bollacker, Lawrence, and Giles 2000). ISI maintains a number of citation databases. It provides two types of search: General Search and Cited Reference Search. ISI only provides simple keyword search. It allows users to find related papers. However, the relevance is judged by the citations that are shared by two papers. That is, if there are one or more common citations between two papers, they are considered as related. However, this method may not reflect the relevance between any two papers accurately, as citation frequency is not considered.

CiteSeer supports the retrieval of scientific literature over the Web. It can automatically locate, parse and index scientific publications found on the Web and generate the citation database. CiteSeer supports two types of keyword search on citations and indexed publications. When searching for citations, all citations matching the given query along with the context of source papers where the citations occur are retrieved. The results are ordered according to the number of times each paper is cited. When searching the full text of indexed publications, CiteSeer returns the header for matching publications along with context of the publication where the keywords occur. Users can order the publications according to the number of citations or by publication date. CiteSeer can also display related publications. The relatedness is calculated using several algorithms. A Term Frequency x Inverse Document Frequency (TFIDF) (Salton and McGill 1983) scheme is used to locate publications with similar words. Distance comparison of publication headers is used to find similar headers. Common Citation x Inverse Document Frequency (CCIDF) (Bollacker, Lawrence, and Giles 1998) is used to find publications with similar citations.

Different from CiteSeer, PubSearch also supports document cluster search apart from the traditional cited reference search. The related publications are grouped into clusters based on common keywords found in their citations. As such, users can retrieve all the related publications even though some publications may not contain the exact keywords users supplied.

WEB CITATION DATABASE

Figure 1 shows the relationship of the two major tables created in the Web Citation Database. They are the SOURCE and CITATION tables. The SOURCE table stores the information of source papers while the CITATION table stores all the citations extracted from the source papers. Most attributes of these two tables have the same data definitions, such as the paper title, author names, journal name, journal volume, journal issue, pages and the year of publication. URL_link is the Web URL address of the corresponding document. With this field, full-text access is possible. “paper_ID” of the SOURCE table and “citation_ID” of the CITATION table are the primary keys in these two tables, respectively. “no_of_citation” of the SOURCE table is the number of references contained in the source paper. “source_ID” of the CITATION table links to the “paper_ID” of the SOURCE table to identify the source paper that cites the particular publication stored in the CITATION table. Most fields in the CITATION table



SOURCE
paper_ID
no_of_citation
paper_title
author1
author2
author3
no_of_authors
affiliation1
affiliation2
affiliation3
journal_name
journal_volume
journal_issue
page_from
page_to
year_of_pub
pub_address
language
URL_link

CITATION
citation_ID
source_ID
paper_title
author1
author2
author3
journal_name
journal_volume
journal_issue
page_from
page_to
year_of_pub
URL_link

FIGURE 1. Database structure of the Web Citation Database.

are similar to those in the SOURCE table. It should also be noted that for all the papers, only the first three authors are stored in the Web Citation Database. This is based on the assumption that the fourth and subsequent authors contribute little to the paper.

An example of records stored in the SOURCE and CITATION tables is illustrated in Figure 2. Records in the SOURCE and CITATION tables have many-to-many relationships. That is, one source paper from the SOURCE table may cite multiple papers in the CITATION table, while one record in the CITATION table may be cited by more than one source paper in the SOURCE table. The example shows that both source papers with paper_ID 1068 and 1124 cite the same paper entitled "A simple blueprint for automatic Boolean query processing" written by Salton. On the other hand, the source paper 1068 cites papers by Salton and by Harter at the same time.

For experimental purpose, we have set up a test Web Citation Database by downloading the publications from 1987 to 1997 in Information Retrieval field of Social Science Citation Index from the Institute for Scientific Information (ISI) Web site, which includes all the journals on Library and Information Science. A total of 1,466 Information Retrieval (IR) related papers were selected from 367 journals with 44,836 citations. The two tables, SOURCE and CITATION, were created based on these IR papers.

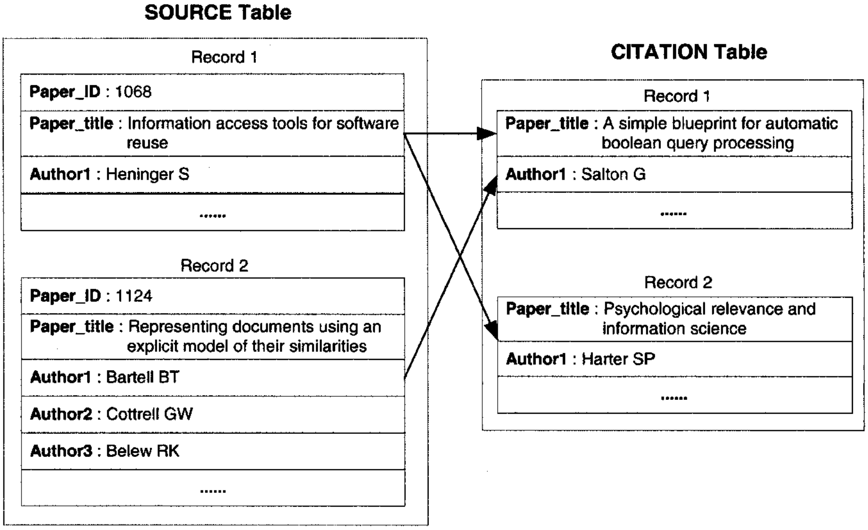

SOURCE Table
Record 1
Paper_ID : 1068
Paper_title : Information access tools for software reuse
Author1 : Heninger S
......
Record 2
Paper_ID : 1124
Paper_title : Representing documents using an explicit model of their similarities
Author1 : Bartell BT
Author2 : Cottrell GW
Author3 : Belew RK
......
CITATION Table
Record 1
Paper_title : A simple blueprint for automatic boolean query processing
Author1 : Salton G
......
Record 2
Paper_title : Psychological relevance and information science
Author1 : Harter SP
......

FIGURE 2. Example of records stored in the SOURCE and CITATION tables.

CITATION DATABASE MINING PROCESS

Figure 3 shows the mining process for document clustering from the Web Citation Database. It consists of five steps, namely feature selection, pre-processing, transformation, document cluster generation, and retrieval. Firstly, the paper titles of the citation records in the Web Citation Database are selected as feature factors to represent the document vectors. Then, the paper titles of citations are pre-processed to extract keywords. The pre-processed keywords are converted into document vectors. In document clusters generation, KSOM and Fuzzy ART are used as the mining techniques to generate document clusters from the document vectors. Finally, the document cluster information is used together with the Web Citation Database to support the retrieval process in the PubSearch system.

Feature Selection

Traditional clustering approaches use TFIDF vector representations for text data (Salton and McGill 1983). Each component of a document vector is calculated as a product of Term Frequency (TF) and Inverse Document Frequency (IDF). The cosine measure can then be used to compute the angle between any two document vectors. Using this method, the documents are classified into different groups according to the distance between them.

TFIDF method is used based on the premise that the full-text of a document is available such that keywords can be extracted as the feature

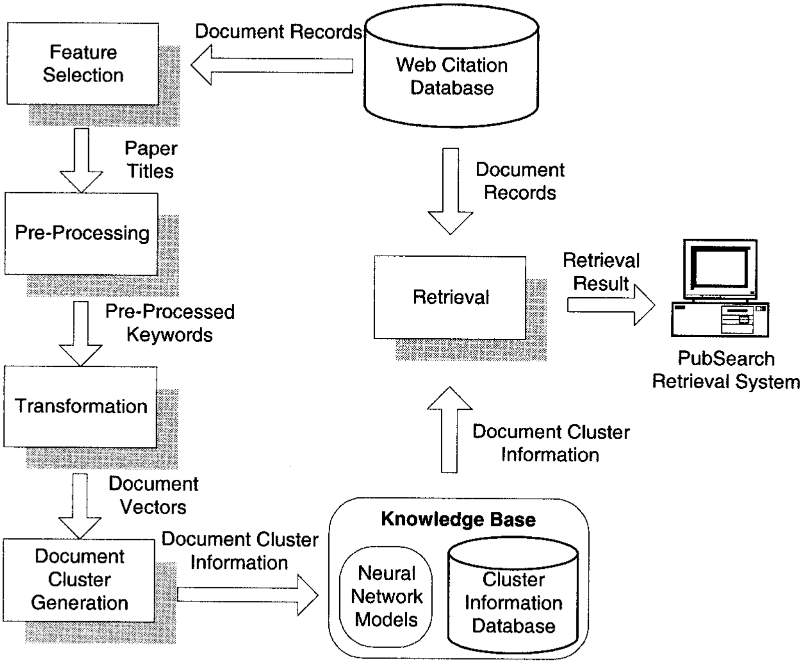
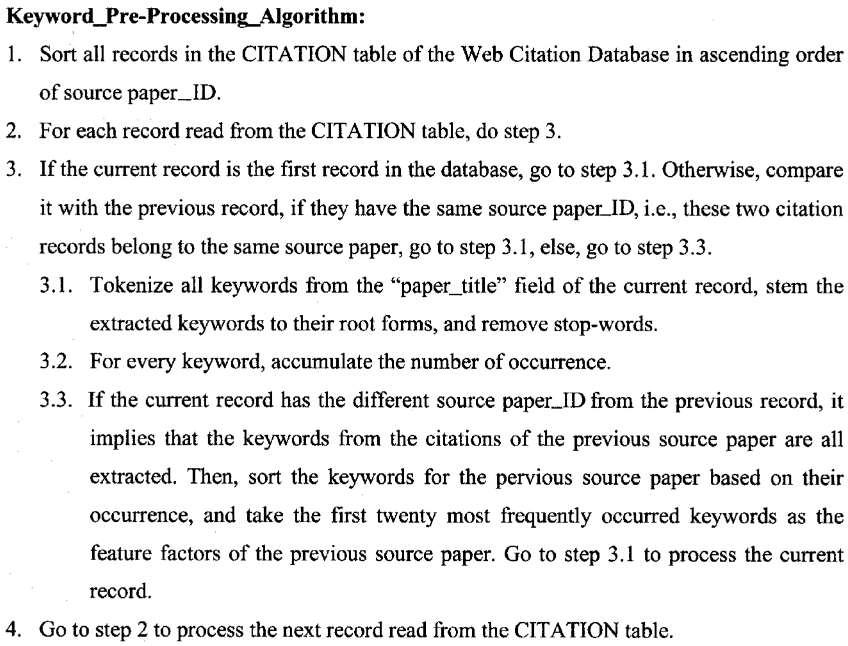


FIGURE 3. Citation database mining process.

factors of the document. However, due to storage limitation, it is not possible to keep the full-text of all source documents into the Web Citation Database. Thus, the traditional approach cannot be adopted here. However, citation information is highly related to the source documents as cited papers are picked as related documents by the authors. Therefore, keywords can be extracted from the titles of all the citations of every document as feature factors. For each document, the twenty most frequently occurred keywords will be extracted from its citations by the *Pre-Processing* step. Then, the TFIDF method can be used to represent the document vector.

Pre-Processing

The *Pre-Processing* step involves text processing techniques, which consist of tokenization, stemming, and stop-word removal. Tokenization breaks the paper titles selected from the *Feature Selection* step into distinct words. Stemming converts the words into their root forms. Stop-word removal removes words with weak or no meaning, such as "to", "the", "a", etc. The WordNet (WordNet 2000) library is used to implement these techniques. The keyword pre-processing algorithm is given in Figure 4. For each source paper, the accumulation of keyword frequency is based on the total number



```
Keyword_Pre-Processing_Algorithm:
1. Sort all records in the CITATION table of the Web Citation Database in ascending order of source paper_ID.
2. For each record read from the CITATION table, do step 3.
3. If the current record is the first record in the database, go to step 3.1. Otherwise, compare it with the previous record, if they have the same source paper_ID, i.e., these two citation records belong to the same source paper, go to step 3.1, else, go to step 3.3.
    3.1. Tokenize all keywords from the "paper_title" field of the current record, stem the extracted keywords to their root forms, and remove stop-words.
    3.2. For every keyword, accumulate the number of occurrence.
    3.3. If the current record has the different source paper_ID from the previous record, it implies that the keywords from the citations of the previous source paper are all extracted. Then, sort the keywords for the pervious source paper based on their occurrence, and take the first twenty most frequently occurred keywords as the feature factors of the previous source paper. Go to step 3.1 to process the current record.
4. Go to step 2 to process the next record read from the CITATION table.
```

FIGURE 4. Keyword pre-processing algorithm.

of the same keyword appearing in the titles of all cited papers. Therefore, the keywords are extracted from the cited paper titles, not the source paper title. Consequently, the algorithm has been developed to handle repetition of keywords. After keyword pre-processing, a total of 5,487 distinct keywords are extracted from all the citations in the Web Citation Database.

Transformation

The *Transformation* step converts documents into vectors before feeding them into the neural networks for training. Traditionally, documents are represented using the vector space model (Salton and McGill 1983) or Latent Semantic Indexing (LSI) (Deerwester et al. 1990). The major drawback of the vector space model is the huge vocabulary in the large collection of free-text documents, which results in a vast dimensionality of the document vectors. LSI tries to reduce the dimensionality of the document vector by incorporating a method called Singular-Value Decomposition (SVD) to omit factors from the document vector that have minimal influence on it. But this method still incurs expensive computation time. In this paper, the random projection method (Kaski 1998) is used to reduce the dimensionality of the document vectors without losing the power of discrimination between documents. The original document vector is multiplied by a random sparse binary projection matrix R that consists of random values. The Euclidean length of each column of R is then normalized to unity.

Kaski (1998) has proved that when the dimension is reduced to *d*, the variance between the document vectors with reduced dimensions and the original document vector is at most $2/d$. If *d* is small, the variance will be large, which results in great loss of the original information. Therefore, *d* needs to be carefully set to retain the original information as much as possible. In this research, a variance between 0.005 and 0.01 is considered acceptable. That is, *d* will be between 200 and 400. We set *d* to 300 in considering the trade-off between the variance and computational complexity. The original document vector is $1 \times 5,487$ and the target document vector is 1×300, therefore, the matrix R should be $5,487 \times 300$. The final document vectors obtained will only have 300 dimensions, which can increase the learning speed dramatically.

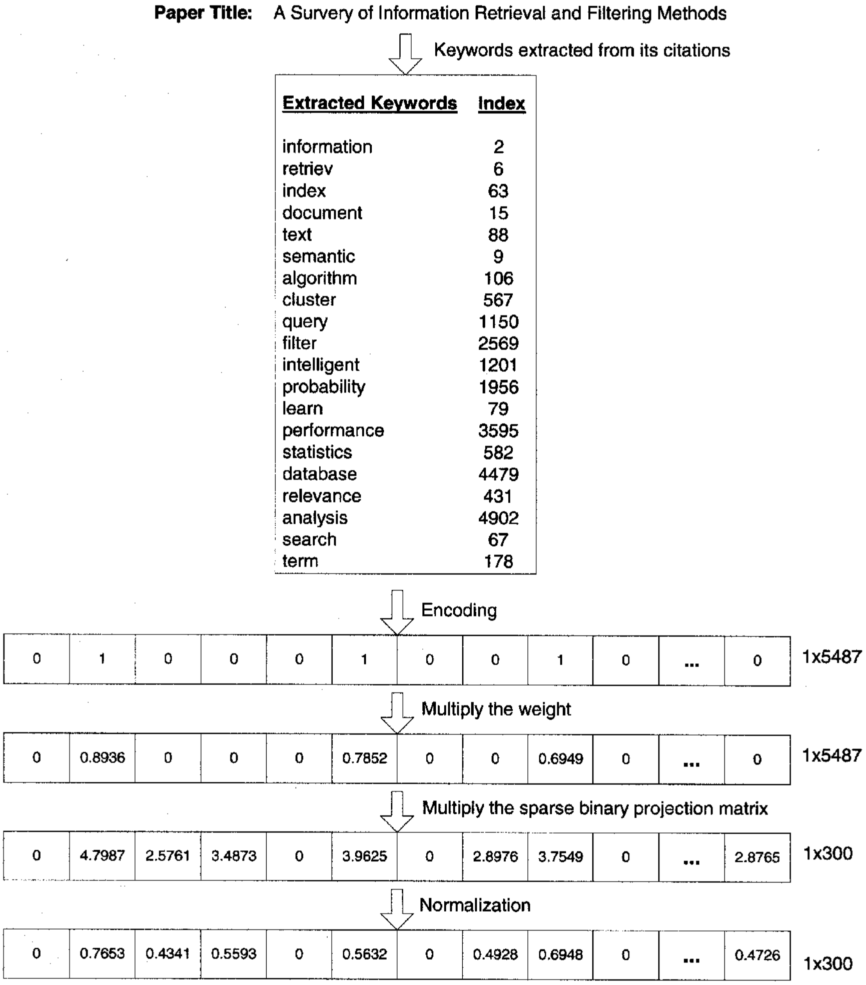
According to Kohonen (1998), a sparse binary projection matrix with exactly five randomly distributed ones in each column is almost as good as the vector space model. An example of the sparse binary projection matrix R is shown in Figure 5. As such, the computational complexity for the matrix product $Y = X \times R$ is O(nd), where *X* is the original document vector, *Y* is the resulting document vector, *n* and *d* are the dimensions before and after the random projection, which are 5,487 and 300 respectively.

An example of the *Transformation* step is illustrated in Figure 6. The first twenty most frequent keywords are extracted from the citations of the source paper. As there are altogether 5,487 distinct keywords extracted from the Web Citation Database, each document will be represented as a vector with 5,487 dimensions. The index column of Figure 6 represents the position of each keyword in the $1 \times 5,487$ vector. The presence of the ith keyword will set the ith element in the document vector to 1. Then, each element of the vector is weighted by TFIDF. The resulting vector is multiplied by the sparse

Sparse Binary Projection Matrix
5,487X300
(each column consists of five ones)

$$\begin{bmatrix} 1 & 0 & 0 & 1 & \dots & 1 \\ 0 & 1 & 1 & 0 & \dots & 0 \\ 1 & 1 & 0 & 0 & \dots & 1 \\ 0 & 0 & 1 & 1 & \dots & 0 \\ 0 & 0 & 1 & 0 & \dots & 1 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 1 & 0 & 0 & \dots & 0 \end{bmatrix}$$

FIGURE 5. An example of the sparse binary projection matrix R.



Paper Title: A Survery of Information Retrieval and Filtering Methods

Extracted Keywords	Index
information	2
retriev	6
index	63
document	15
text	88
semantic	9
algorithm	106
cluster	567
query	1150
filter	2569
intelligent	1201
probability	1956
learn	79
performance	3595
statistics	582
database	4479
relevance	431
analysis	4902
search	67
term	178

0	1	0	0	0	1	0	0	1	0	...	0	1x5487

0	0.8936	0	0	0	0.7852	0	0	0.6949	0	...	0	1x5487

0	4.7987	2.5761	3.4873	0	3.9625	0	2.8976	3.7549	0	...	2.8765	1x300

0	0.7653	0.4341	0.5593	0	0.5632	0	0.4928	0.6948	0	...	0.4726	1x300

FIGURE 6. An example of the *Transformation* step.

binary projection matrix to reduce its dimension from 5,487 to 300. Finally, the vector is normalized before feeding into the neural network.

Document Cluster Generation

The mining techniques used for document cluster generation are the KSOM (Kohonen 1995) and Fuzzy ART (Carpenter, Grossberg, and Rosen 1991) neural networks. To categorize the source documents into different clusters, training is needed for both neural networks. The training procedures for KSOM and Fuzzy ART are different due to the differences in the architectures of these two models.

For KSOM neural network training, the weights of the network are first initialized with random real numbers within the interval [0,1]. A total of 1,000 records in the SOURCE table are used as the training set. The performance of the KSOM neural network retrieval depends on the number of clusters generated (i.e. the number of neurons in the network), and the average number of documents within a cluster. However, decision on the best size of the cluster map remains a non-trivial problem that requires some insight into the structure of the training data. In our implementation, the number of clusters to be generated is set to 100 in order to obtain fast retrieval speed. The initial neighborhood size is set to half the number of the clusters. The number of iterations and the initial learning rate are set to 5,000 and 0.5, respectively. During the training, whenever a winner cluster is found, the weights to the winner cluster together with its neighborhood need to be updated according to the input pattern. The updated weights are then stored. During retrieval, no weight updates will be performed. This prevents any incoming query from corrupting the "index" stored in the neural networks.

A Fuzzy ART system includes a pre-processing field of nodes F_0, an input field F_1, and a category representation field F_2. F_0 modifies the current input vector, while F_1 receives both bottom-up input from F_0 and top-down input from F_2. Three parameters are used to determine the dynamics of a Fuzzy ART network, namely a choice parameter $\alpha > 0$, a learning rate parameter $\beta \in [0, 1]$, and a vigilance parameter $\rho \in [0, 1]$. The choice parameter α affects the bottom-up inputs that are produced at the F_2 nodes according to the input patterns presented at F_1. As short training time is only possible for small values of the choice parameter (Carpenter et al. 1991), α is set to 0.2 in our work. β controls the adjustment of the weight vector W_j. We set $\beta = 1$ if j is an uncommitted node and $\beta = 0.5$ if j is a committed node. The vigilance threshold level indicates how close an input must be to a stored cluster to provide a desirable match. The higher the vigilance threshold, the more precise the documents are clustered. However, in order to compare the performance of KSOM and Fuzzy ART more closely, the number of clusters to be generated using Fuzzy ART is set as close as possible to the number obtained in KSOM, which is 100. Therefore, the vigilance threshold is set to 0.7 in Fuzzy ART, which has generated 129 clusters.

Retrieval

A user submits a query for publication retrieval during the *Retrieval* step. The query is then pre-processed, parsed and encoded in a similar way as the *Pre-Processing* and *Transformation* steps. As discussed in the previous section, a total of 5,487 distinct keywords are extracted from all the citations in the Web Citation Database. The input keywords are compared with these 5,487 words to form a 5,487-dimensional query vector. The length of the vector is fixed, as it is fed into the fixed number of input neurons of the neural network. Thus, every element of the vector must have an assigned value. The occurrence of the keyword will set the element of the query vector in the corresponding position to one. Otherwise, the keyword will be used as the index term to read from the WordNet thesaurus to find its synonyms. If any of the synonyms is found in the 5,487-keyword list, the original keyword in the query is replaced by that synonym. If it fails to find the matched keywords from the WordNet thesaurus, that element of the query vector will be set to 0. This has the effect of omitting the corresponding term while maintaining the overall length of the vector.

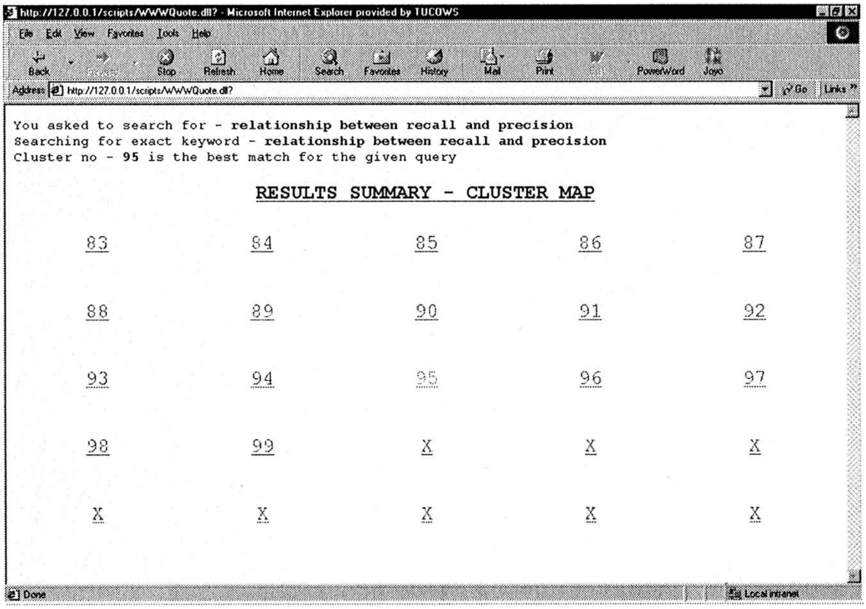
During the weight multiplication step, instead of using the weight term TFIDF as used in the document vector, each element in the query vector is weighted by $qf \times idf$ (the frequency of the term in the query × the inverse document frequency of the term in the collection) (Turtle and Croft 1991). This is based on the assumptions that a content-bearing term that occurs frequently in the query is more likely to be important than one that occurs infrequently, and terms that occur infrequently in the document collection are more likely to be important than frequent or common terms.

The encoded user query items are fed into the network to determine which clusters to be activated. The documents in the activated cluster are ranked according to the Euclidean distance to the query term (Salton 1991). Given a document vector d and a query vector q, their similarity $sim(d, q)$ is given as follows:

$$sim(d, q) = \frac{\sum_{i=1}^{n} (w_{di} \times w_{qi})}{\sqrt{\sum_{i=1}^{n} (w_{di})^2 \times \sum_{i=1}^{n} (w_{qi})^2}}$$

where w_{di} and w_{qi} are the weight of the ith element in the document vector d and query vector q respectively. The document having the minimum value of $sim(d, q)$ in a cluster is given the highest ranking within that cluster. Other documents in the cluster are sorted based on this principle. In the event that two or more documents in the cluster have the same $sim(d, q)$ value, the ranking among these documents is done randomly.

Figure 7 shows the cluster map for the query "relationship between recall and precision" using the KSOM neural network. This example illustrates that semantic clusters are spread across several units. The best-matched cluster "95" is highlighted, which is displayed together with its neighborhood clusters. Thus, users can explore documents resided in the neighboring clusters in addition to the best-matched cluster. The client Web interface allows the user to browse through the cluster map. By clicking any of the cluster numbers in the cluster map, documents from that particular cluster are listed and ranked according to the least Euclidean distance. This is shown


http://127.0.0.1/scripts/WWWQuote.dll? - Microsoft Internet Explorer provided by TUCOWS
File Edit View Favorites Tools Help
Back
Forward
Stop
Refresh
Home
Search
Favorites
History
Mail
Print
PowerWord
Jojo
Address http://127.0.0.1/scripts/WWWQuote.dll?
Go
Links
Done
Local intranet

```
You asked to search for - relationship between recall and precision
Searching for exact keyword - relationship between recall and precision
Cluster no - 95 is the best match for the given query
```

RESULTS SUMMARY - CLUSTER MAP

83	84	85	86	87
88	89	90	91	92
93	94	95	96	97
98	99	X	X	X
X	X	X	X	X

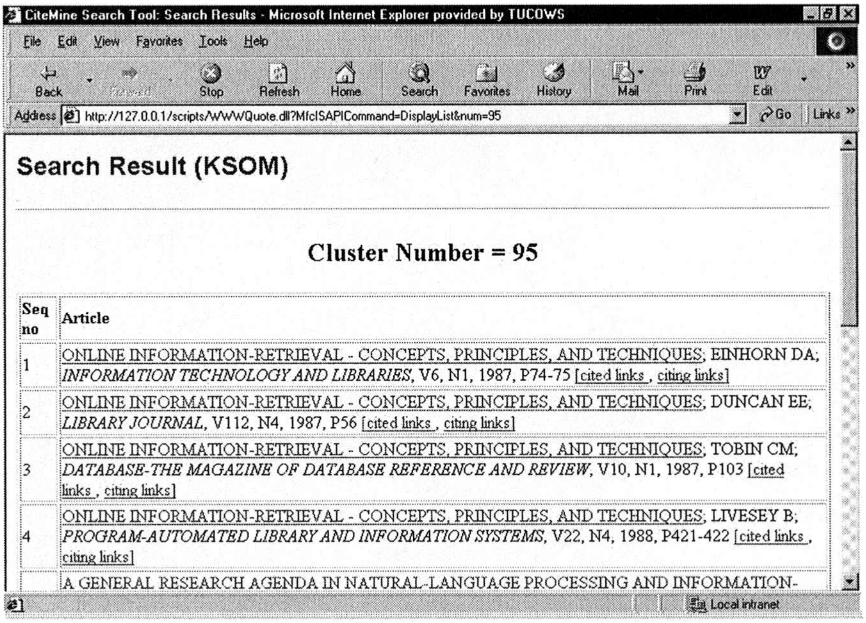
FIGURE 7. Cluster map for the KSOM algorithm.

in Figure 8. The paper titles are underlined to allow the user to get the full-text content of the paper through the underlying URL links. There are also "citing" and "cited" links provided, which allow the user to go deeper into the citing or cited documents of that particular publication.

Retrieval of the stored documents in the Fuzzy ART neural network is identical to the training process. However, the vigilance test is always passed and no weight updates are performed during retrieval. Figure 9 shows the search result for the same query "relationship between recall and precision" using the Fuzzy ART network. The total number of documents returned is 87, which is much greater than the total number of documents returned using the KSOM network, which has 32. Another difference from KSOM is that there is no neighborhood clusters information displayed using Fuzzy ART. This is because Fuzzy ART network is sensitive to the sequence of the training data. In another words, the training data presented to the Fuzzy ART network in different sequence will result in different clusters being generated. Therefore, there is no direct relationship between the winning cluster and its neighborhood clusters.

PERFORMANCE EVALUATION

The PubSearch citation-based retrieval system has been developed. In this section, we present performance evaluation of the system. As PubSearch

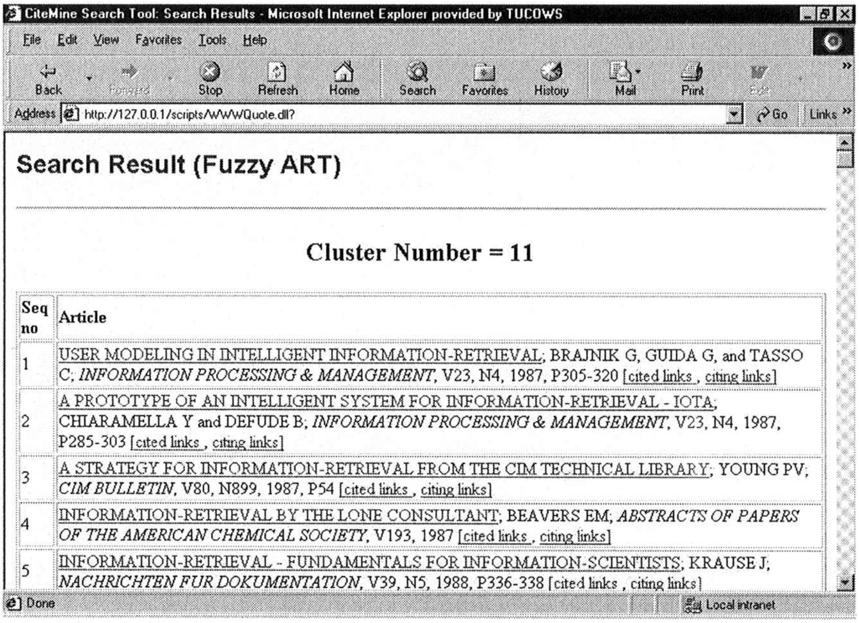

CiteMine Search Tool: Search Results - Microsoft Internet Explorer provided by TUCOWS
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites History Mail Print Edit
Address http://127.0.0.1/scripts/WWWQuote.dll?MfcISAPICommand=DisplayList&num=95
Go Links
Local intranet

Search Result (KSOM)

Cluster Number = 95

Seq no	Article
1	ONLINE INFORMATION-RETRIEVAL - CONCEPTS, PRINCIPLES, AND TECHNIQUES; EINHORN DA; *INFORMATION TECHNOLOGY AND LIBRARIES*, V6, N1, 1987, P74-75 [cited links , citing links]
2	ONLINE INFORMATION-RETRIEVAL - CONCEPTS, PRINCIPLES, AND TECHNIQUES; DUNCAN EE; *LIBRARY JOURNAL*, V112, N4, 1987, P56 [cited links , citing links]
3	ONLINE INFORMATION-RETRIEVAL - CONCEPTS, PRINCIPLES, AND TECHNIQUES; TOBIN CM; *DATABASE-THE MAGAZINE OF DATABASE REFERENCE AND REVIEW*, V10, N1, 1987, P103 [cited links , citing links]
4	ONLINE INFORMATION-RETRIEVAL - CONCEPTS, PRINCIPLES, AND TECHNIQUES; LIVESEY B; *PROGRAM-AUTOMATED LIBRARY AND INFORMATION SYSTEMS*, V22, N4, 1988, P421-422 [cited links , citing links]
	A GENERAL RESEARCH AGENDA IN NATURAL-LANGUAGE PROCESSING AND INFORMATION-

FIGURE 8. The result for the cluster number 95 using the KSOM algorithm.


CiteMine Search Tool: Search Results - Microsoft Internet Explorer provided by TUCOWS
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites History Mail Print Edit
Address http://127.0.0.1/scripts/WWWQuote.dll?
Go Links
Done
Local intranet

Search Result (Fuzzy ART)

Cluster Number = 11

Seq no	Article
1	USER MODELING IN INTELLIGENT INFORMATION-RETRIEVAL; BRAJNIK G, GUIDA G, and TASSO C; *INFORMATION PROCESSING & MANAGEMENT*, V23, N4, 1987, P305-320 [cited links , citing links]
2	A PROTOTYPE OF AN INTELLIGENT SYSTEM FOR INFORMATION-RETRIEVAL - IOTA; CHIARAMELLA Y and DEFUDE B; *INFORMATION PROCESSING & MANAGEMENT*, V23, N4, 1987, P285-303 [cited links , citing links]
3	A STRATEGY FOR INFORMATION-RETRIEVAL FROM THE CIM TECHNICAL LIBRARY; YOUNG PV; *CIM BULLETIN*, V80, N899, 1987, P54 [cited links , citing links]
4	INFORMATION-RETRIEVAL BY THE LONE CONSULTANT; BEAVERS EM; *ABSTRACTS OF PAPERS OF THE AMERICAN CHEMICAL SOCIETY*, V193, 1987 [cited links , citing links]
5	INFORMATION-RETRIEVAL - FUNDAMENTALS FOR INFORMATION-SCIENTISTS; KRAUSE J; *NACHRICHTEN FUR DOKUMENTATION*, V39, N5, 1988, P336-338 [cited links , citing links]

FIGURE 9. Search result of the Fuzzy ART network.

is based on the mining of the Web Citation Database using KSOM and Fuzzy Art techniques for document clustering, performance is conducted on both the KSOM and Fuzzy ART neural networks. We compare both the training performance and retrieval performance of these two models. The experiments were carried out on a Pentium II 450 MHz machine with 128M RAM under the Windows NT operating system.

Training Performance

In this experiment, the following data are used. The number of keywords in the keyword list is 5,487. The number of words to be searched in the WordNet dictionary is 121,962. The total number of documents used for training (the training set) is 1,000. The training performance is evaluated in two aspects, one is training efficiency and the other is training accuracy. Training efficiency is measured based on the total training time and the number of the iterations required by the neural networks to reach the convergent state. Table 1 shows the training performance of these two neural networks.

Training accuracy is measured by evaluating the effectiveness of cluster assignments. The standard recall, precision and F_1 measure are used. Recall is defined as the ratio of correct assignments by the system divided by the total number of correct assignments. Precision is the ratio of correct assignments by the systems divided by the toal number of system's assignments. The F_1 measure, which was introduced by Van Rijsbergen (1979), combines recall (r) and precision (p) with an equal weight as follows:

$$F_1(r,p) = \left(\frac{2rp}{r+p}\right)$$

The F_1 scores can be calculated for each cluster and averaged across the experiments. Two kinds of averaging methods can be used: micro-averaging and macro-averaging techniques (Yang and Liu 1999). Micro-averaging scores are computed on a per-document basis. They tend to be dominated by the system's performance on large clusters. Macro- averaging scores are computed on a per-cluster basis. Therefore, they are more likely to be

TABLE 1 Training Efficiency of KSOM and Fuzzy ART

Criteria	KSOM	Fuzzy ART
Pre-processing time	1 min 6 sec	1 min 6 sec
No. of iterations	5,000	800
Training time	46 min 25 sec	12 min 11 sec
Total no. of clusters	100	129

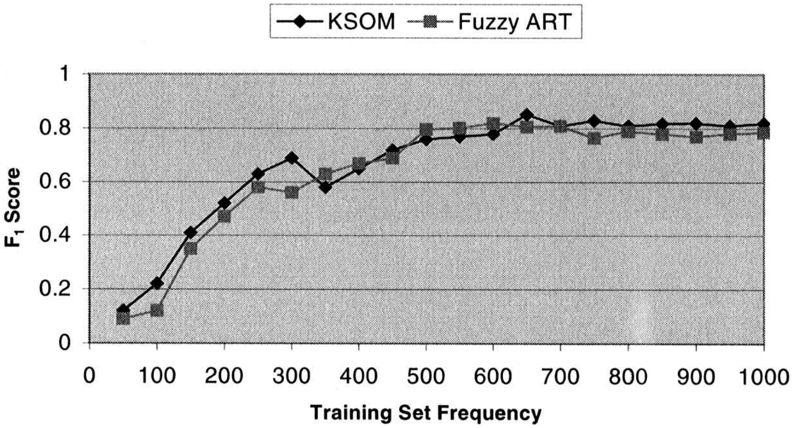


FIGURE 10. Performance of training accuracy for KSOM and Fuzzy ART.

influenced by the system's performance on small clusters. We only measure micro-averaging F_1 in the performance evaluation as it has been widely used in cross-method comparison (Yang and Liu 1999). That is, the F_1 value is computed globally over all the $n \times m$ binary decisions, where n is the number of total training documents, and m is the number of clusters.

The precision and recall for each cluster have been calculated and used to derive the F_1 measure, which reflects the overall system performance. Figure 10 shows the F_1 scores for both KSOM and Fuzzy ART. The horizontal axis is divided into equal-sized intervals for the training set frequency ranging from 50 to 1,000. The vertical axis represents the F_1 score. The curves are obtained by averaging the per-cluster F_1 scores per interval for each neural network algorithm and interpolating the F_1 scores. Although at some points, Fuzzy ART performs better than KSOM, the overall performance of KSOM is still better than Fuzzy ART. Also, the F_1 scores become stable when the training set is large.

In addition, the characteristics of document distributions for the training set with size 1,000 were measured. The average number of documents per unit depends on the input samples. Different document samples will give different results. In this research, we measured the average number of documents contained in the largest and smallest clusters together with the corresponding standard deviation for KSOM and Fuzzy ART neural networks. The results are given in Table 2.

Retrieval Performance

The retrieval performance is measured based on the average online retrieval speed and retrieval precision. Retrieval speed measures how fast

TABLE 2 Characteristics of Document Distributions of KSOM and Fuzzy ART

	Average Number of Documents		Standard Deviation	
Technique	Largest Cluster	Smallest Cluster	Largest Cluster	Smallest Cluster
KSOM	27	5	0.72	0.53
Fuzzy ART	35	3	0.81	0.49

the result is presented to the user after submitting a search query. Retrieval precision is measured as the ratio of the number of documents that are judged as relevant for a particular query over the total number of documents retrieved. In this research, both system-based and user-based relevance measures (Harter 1992; Saracevic 1996) are used. The system-based relevance measurement is carried out directly by examining the ranked list of documents generated by the system, while the user-based relevance measurement is based on the judgements of the users on the relationship between a query representation and a retrieved document. In (Pao 1993), user based relevant assessments are made according to three categories: highly relevant, partially relevant, and not relevant. In this research, we have introduced two more categories for better refinement. The five categories with values of 1.0, 0.75, 0.5, 0.25 and 0, respectively, are used in the assessment.

The experiment was conducted as follows. Fifty queries were submitted to the system, the average online retrieval speed was measured from the time difference between the moment of query submission and the point when results were displayed to the user. For each query result, only the first 20 documents were examined to check the relevance to the query. The user-based relevance is measured from an average of 10 users. Table 3 gives the retrieval performance of the KSOM and Fuzzy ART neural networks. It can be observed that KSOM has slower retrieval speed but with higher overall retrieval precision. The standard deviation of the KSOM neural network is also smaller than the Fuzzy ART network.

TABLE 3 Retrieval Performance of KSOM and Fuzzy ART

		Retrieval Precision			
Technique	Average Retrieval Speed	System-Based Relevance	User-Based Relevance	Average	Standard Deviation
KSOM	1.6 sec	84.35%	78.5%	81.43%	0.24
Fuzzy ART	0.7 sec	83.12%	71.25%	77.19%	0.38

CONCLUSION

The World Wide Web has become an important medium for disseminating scientific publications. In this paper, we have proposed a mining process to extract document cluster knowledge from the Web Citation Database that can be used to support the retrieval of scientific publications over the Web. The citation database mining process consists of five steps: feature extraction, pre-processing, transformation, document cluster generation, and retrieval. The mining techniques used for document cluster generation are based on the KSOM and Fuzzy ART neural networks.

Performance evaluation has also been conducted for the two mining techniques. The results have shown that both the KSOM and Fuzzy ART networks are capable to uncover the semantic similarities of documents based on the feature vector representation of the documents. The KSOM network has achieved better retrieval precision, but at the expense of longer training time. However, Fuzzy ART is more suited to the Web publication environment as the Web Citation Database is dynamic with frequent updates. The clustering results have been incorporated into the PubSearch retrieval system to support retrieval of Web scientific publications. As the Web Citation Database also contains other useful information, we are currently investigating other mining techniques for author co-citation analysis (White and Griffith 1981) and co-word analysis (Callon et al. 1991) to support publication retrieval.

REFERENCES

Aggarwal, C., S. Gates, and P. Yu. 1999. On the merits of building categorization system by supervised clustering. *Proceedings of the 5th ACM SIGKDD International Conference on knowledge Discovery and Data Mining*, 15–18 August 1999, San Diego, CA, 352–356.

Bollacker, K., S. Lawrence, and C. Giles. 1998. CiteSeer: an autonomous Web agent for automatic retrieval and identification of interesting publications. *Proceedings of the 3rd ACM Conference on Digital Libraries*, 23–26 June 1998, Pittsburgh, PA, 116–123.

Bollacker, K., S. Lawrence, and C. Giles. 2000. Discovering relevant scientific literature on the Web. *IEEE Intelligent Systems* 15(2): 42–47.

Callon, M., J. P. Courtial, and F. Laville. 1991. Co-word analysis as a tool for describing the network of interactions between basic and technological research: the case of polymer chemistry. *Scientometrics* 22(1): 153–203.

Carpenter, G., S. Grossberg, and D. Rosen. 1991. Fuzzy ART: fast stable learning and categorization of analog patterns by an adaptive resonance system. *Neural Networks* 4: 759–771.

Deerwester, S., S. Dumais, G. Furnas, and K. Landauer. 1990. Indexing by latent semantic analysis. *Journal of American Society for Information Science* 41: 391–407.

Digital Equipment Corporation (DEC). 2000. *Virtual Paper Project*. http://www/research.digital.com/SRC/virtualpaper/home.html.

Fayyad, U. M., G. Piatetsky-Shapiro, and P. Smyth. 1996. From data mining to knowledge discovery: an overview. U. M., Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy. (eds.) *Advances in Knowledge Discovery and Data Mining*, 1–34. Menlo Park, CA: AAAI/MIT Press.

Garfield, B. 1979. *Citation Indexing: Its Theory and Application in Science, Technology, and Humanities*. New York: John Wiley & Sons.

Harter, S. P. 1992. Psychological relevance and information science. *Journal of the American Society for Information Science* 43: 602–615.

Hartigan, J. A. 1975. *Clustering Algorithms*. New York: John Wiley and Sons.

He, Y. 2000. Mining a Web Citation Database for the Retrieval of Scientific Publications over the WWW. M.A.Sc. Thesis, School of Computer Engineering, Nanyang Technological University, Singapore.

Honkela, T., S. Kaski, T. Kohonen, and K. Lagus. 1998. Self-organizing maps of very large document collections: Justification for the WEBSOM method. L. Balderjahn, R. Mathar, and M. Schader. (eds.) *Classification, Data Analysis, and Data Highways*. 245–252. Berlin: Springer.

Institute for Scientific Information (ISI). 2000. http://www.isinet.com.

Jain A. K., M. N. Murty, and P. J. Flynn. 1999. Data clustering: a review. *ACM Computer Surveys* 31(3): 264–323.

Kaski, S. 1998. Dimensionality reduction by random mapping: fast similarity computation for clustering. *Proceedings of International Joint Conference on Neural Networks (IJCNN'98)* 5–9 May 1998, Anchorage, AK, 1: 413–418.

Kaski, S., K. Lagus, T. Honkela, and T. Kohonen. 1998. Statistical aspects of the WEBSOM system in organizing document collections. *Computing Science and Statistics* 29: 281–290.

Kaufman, L. and P. Rousseeuw. 1990. *Finding Groups on Data: An Introduction to Cluster Analysis*. New York: John Wiley and Sons.

Kohonen, T. 1995. *Self-Organizing Maps*. Springer.

Kohonen, T. 1998. Self-organizing of very large document collections: state of the art. *Proceedings of the 8th International Conference on Artificial Neural Networks*, 2–4 September 1998, Skövde, Sweden, 65–74.

Kohonen, T., S. Kaski, K. Lagus, J. Salojärvi, V. Paatero, and A. Saarela. 2000. Self organization of a massive document collection. *IEEE Transactions on Neural Networks*. Special Issue on Neural Networks for Data Mining and Knowledge Discovery, 11(3): 574–585.

Lin, X. 1997. Map displays for information retrieval. *Journal of the American Society for Information Science* 48: 40–54.

Mitchell, T. 1999. Machine learning and data mining. *Communications of the ACM* 42(11): 31–36.

Pao, M. L. 1993. Term and citation retrieval: a field study. *Information Processing & Management* 29(1): 95–112.

Rauber, A., and D. Merkl. 1999. SOMLib: A digital library system based on neural networks. *Proceedings of the 4th ACM Conference on Digital Libraries (DL'99)*, 11–14 August, 1999, Berkeley, CA, 240–241.

Rocchio, J. 1966. Document Retrieval Systems–Optimization and Evaluation: Ph.D. Diff Harvard University, Cambridge, MA USA.

Salton, G. 1991. Developments in automatic text retreival. *Science* 253: 974–979.

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

Saracevic, T. 1996. Relevance reconsidered. P. Ingwersen and N. O. Pors (Eds.) *Information Science: Integration in Perspective*. Copenhagen: Royal School of Librarianship, 210–218.

Schatz, B., and H. Chen. 1996. Building large-scale digital libraries. *IEEE Computer* 29(5): 22–26.

Slonim, N., and N. Tishby. 2000. Document clustering using word clusters via the information bottleneck method. *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 24–28 July 2000, Athens, Greece, 208–215.

Tishby, N., F. C. Pererira, and W. Bialek, 1999. The information bottleneck method. *Proceedings of 37th Allerton Conference on Communication and Computation*, 22–24 September 1999, Urbana, IL, 368–377.

Turtle, H., and W. B. Croft. 1991. Evaluation of an inference network-based retrieval model. *ACM Transactions on Information Systems* 9(3): 187–222.

Van Rijsbergen, C. 1979. *Information Retrieval. 2nd ed.* London, England: Utterworths.

White, H. D., and B. C. Griffith. 1981. Author co-citation: a literature measure of intellectual structure. *Journal of the American Society for Information Studies* 32: 163–171.

WordNet. 2000. *WordNet-A Lexical Database for English*. http://www.cogsci.princeton.edu/~wn.

Yang, Y., and X. Liu. 1999. A re-examination of text categorization methods. *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 15–19 August 1999, Berkeley, CA, 42–49.

Zamir, O., and O. Etzioni, 1998. Web document clustering: a feasibility demonstration. *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 24–28 August 1998, Melbourne, Australia, 46–54.