Journal of Computer Science 2 (5): 434-440, 2006
ISSN 1549-3636


# A New Filtering Algorithm for Duplicate Document Based on Concept Analysis

Ahmad M. Hasnah
University of Qatar, Department of Computer Science, Doha, Qatar, P.O. Box 2713

**Abstract:** Data bases and web pages contain currently a huge number of duplicate document. It is then fundamental to have a filter which can be embedded, for instance, within an information retrieval system like a search engine in order to prohibit the redundant documents references to appear on the screen as a reply to the user's query. This filter can save the user time and increases his satisfaction. In this study, we propose a new algorithm based on concept analysis principle, which can act as a filter for duplicate document. It can be applied on a collection of documents or databases and reduce their storage spaces by eliminating redundant documents without loosing knowledge. Our experiments show that this algorithm increases the precision of the information retrieval system and improves its performance.

**Key words:** Duplicate document, concept analysis, information retrieval, information filtering

## INTRODUCTION

Computer machines spread nowadays overall the world and contain a huge number of data of different types such as text documents, databases, images, etc. These computers form a network of networks called the Internet and their data, which can be duplicated, become shared and accessible from different locations. The major problem is that when searching for information over this network, we get a large number of references to redundant data where exploring all these references is extremely hard and may lose the user time and effort. Thus finding a filter that can eliminate the duplicate data references is very important and can improve dramatically the performance and precision of search engines. In addition, this filter can be used to eliminate redundant documents from a collection of data and thus saving on the storage space. The detection of duplicate document within a collection of data has become nowadays an area of research of great interest and many works have been published recently in this field[1-5]. A problem introduced by the indexing of duplicate document is potentially skewed collection statistics. Collection statistics are often used as a part of the similarity computation of a query to a document. Hence, the biasing of the collection statistics may affect the overall precision of the entire system. Simply but, not only is a given user's retrieval performance compromised by the existence of duplicates, but also the overall retrieval accuracy of the engine is likewise jeopardized.

In this study, we propose a new algorithm of data reduction that can eliminate the references to redundant data to appear on the screen as a reply to users queries. It can be used also, to reduce the size of stored information in databases or collections of documents, to keep only the significant data without losing knowledge. In addition, it can be applied on expert systems to reduce the number of production rules stored in the system knowledge database with the ability to regenerate the original set of rules when needed. This algorithm is based on formal concept analysis, which has been developed by several researchers in the world for different scientific applications. In fact, Ganter and Wille[6] gave the mathematical foundation of the formal concept analysis, based on lattice theory. Other research teams in Canada[7] and in Tunis (Jaoua, Ounally, Ben Yahia and Elloumi[8-10], have applied formal concept analysis for *supervised learning*, *information engineering* and *data organization*.

**Document detection techniques are partitioned into three main categories:** *shingling techniques, similarity measures calculations and document images*. Shingling techniques, such as COPS[11], KOALA[12] and DSC[1], take a set of contiguous terms (or shingles) of documents and compare the number of matching shingles. The comparison of document subsets allows the used algorithms to calculate the percentage of overlap between the documents. This type of approach relies on hash values for each document subsection and filters those hash values to reduce the number of comparisons. In the shingling approach, subdocuments are compared instead of comparing full documents, thus, each document produces many potential duplicates. Returning many potential matches requires vast user involvement to sort out potential duplicates, diluting the potential usefulness of the approach. The third approach that computes document-to-document similarity measures[13-15], is similar to document clustering work[16] in that it uses similarity computations to group potentially duplicate document. All pairs of documents are compared, each document is compared to every other document and a similarity

**Corresponding Author:** Ahmad M. Hasnah, University of Qatar, Department of Computer Science, Doha, Qatar, P.O. Box 2713

measure is then calculated. A document to document similarity comparison approach is thus computationally prohibitive given the theoretical $O(d^2)$ runtime, where $d$ is the number of documents.

We mention that, there are other techniques used for images duplicate detection which are detailed in[17,18]. These approaches are specific for images processing rather than documents processing, therefore they are not discussed in this study.

**Mathematical foundations of concepts analysis:** Among the mathematical theories found recently with important applications in computer science, lattice theory has a specific place for data organization, information engineering and data mining for reasoning. It may be considered as the mathematical tool that unifies data and knowledge (or information retrieval and reasoning[19]). In this section, we define the binary context, the formal concept and the lattice of concepts associated with the binary context.

**Definition 1 (binary context):** A *binary context* (or *binary relation*) is a subset of the product of two sets $O$ (set of objects) and $P$ (set of properties).

**Example 1:** Let $O$ ={*Leech, Bream, Frog, Dog, Spike-weed, Reed, Bean, Maize*} and let $P$ = {$a$, $b$, $c$, $d$, $e$, $f$, $g$, $h$, $i$}, where $O$ is a set of some animals and $P$ the set of the following properties:

| | | |
|---|---|---|
| a = needs water | b = lives in water | c = lives on land |
| d = needs chlorophyll toproduce food | e = is two seed leaves | f = One seed leaf |
| g = Can move around | h = has limbs | i = suckles its offspring |

A binary context $R$, may be defined by the following table presenting a binary relation:

| | a | b | c | d | e | f | g | h | i |
|---|---|---|---|---|---|---|---|---|---|
| Leech | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| Bream | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |
| Frog | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 |
| Dog | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 |
| Spike-Weed | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 |
| Reed | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 |
| Bean | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| Maize | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 |

Let $f$ be a function from the powerset of the set of objects $O$ ($2^O$ ) into the powerset of the set of properties $P$ ($2^P$), such that:

$f(A) = \{m \mid g \in A \Rightarrow (g,m) \in R$

$f(A)$ is the set of properties shared by all objects of $A$ (subset of $O$); and $g$ a function from $2^P$ to $2^O$, such that: $g(B) = \{g \mid m \in B \Rightarrow (g,m) \in R\}$; $g(B)$ is the set of objects sharing all the properties $B$ (subset of $P$). We also define the following relation:

closure($A$) = $g(f(A)) = A'$ and closure($B$) = $f(g(B)) = B'$

The meaning of $A'$ is that a set of objects $A$ is sharing the same set of properties $f(A)$ with other objects ($A'$-$A$), relatively to the context $R$. $A'$ is the maximal set of objects sharing the same properties as objects $A$. In example 1, $A$ = {*Leech, Bream, Frog, Spike-weed*} then $A'$={*Leech, Bream, Frog, Spike-weed, Reed*}. This means that the shared properties $a$ and $b$ of animals in $A$, are also shared by a *Reed*, the only element in $A'$-$A$. The meaning of $B'$ is that if an object $x$ of the context $R$, verifies properties $B$, then $x$ verifies also some number of additional properties ($B'$-$B$). $B'$ is the maximal set of properties shared by all objects verifying properties $B$. In example 1, if $B$ ={$a$,$h$}, then $B'$={$a$,$h$,$g$}. This means that any animal that needs water ($a$) and has lambs ($h$), can move around ($g$). For each subset $B$, we may create an association rule $B \rightarrow B'$-$B$. The number of these rules depends on the binary context $R$. In[6], we can find different algorithms to extract the minimal set of such association rules.

**Definition 2 (formal concept):** A *formal concept of binary context*, is the pair ($A$,$B$), such that $f(A)$=$B$ and $g(B)$=$A$. We call $A$ the extent and $B$ the intent of the concept ($A$,$B$).

**Lattice of concepts:** From a context $R$, we can extract all possible concepts. In[6], It is proven that the set of all concepts may be organized as a lattice, when defining the following order relation << between two concepts:
($A1$,$B1$) << ($A2$,$B2$) $\Leftrightarrow$ ($A1 \subseteq A2$) and ($B2 \subseteq B1$).

**Duplicate document reduction:** The main goal of document reduction technique is to remove redundant documents from the documents collection or from the retrieved set of documents. Redundant documents are defined to be duplicate document and documents with information contained by other set of documents. Removing redundancy from a collection of documents will save the storage space and improve the performance of the retrieval system. Eliminating redundant documents from the retrieved set of documents saves the user time and improves the retrieval system precision. In the vector space model for information retrieval system, each document is represented by a vector in the $n$-dimensional space, where each dimension represents an index term. If a binary weighting mechanism is used, a vector of zeros and ones represents each document. The value zero means that the index term does not belong to the document representation and the value one means that the index term is part of the document representation. As a result, a table of zeros and ones represents the document collection (a binary context, Fig. 1). For the purpose of document reduction, we prove that some rows (documents) may be removed from the initial collection (binary context) without losing knowledge. We need to define an equivalence relation between a document and a set of documents. We introduce an exact solution for documents reduction in the case of documents binary representation.

| | T1 | T2 | T3 | T4 | ..... | Tn |
|---|---|---|---|---|---|---|
| DOC 1 | 1 | 0 | 1 | 1 | ..... | 0 |
| DOC 2 | 0 | 0 | 1 | 0 | ..... | 1 |
| DOC 3 | 1 | 1 | 0 | 1 | ..... | 1 |
| . . . | | | | | | |
| DOC n | 0 | 0 | 1 | 0 | ..... | 0 |

Fig. 1: Documents representations (a binary context)

**Equivalence between a document/object and a subset of documents/objects:** We say that a document $D_i$ is equivalent to a set of documents $S_D$, relatively to a binary context $R$, if and only if, $\{D_i\}$ U $S_D$ is a domain of a concept of $R$ and that the closure($D_i$) = closure($S_D$) = $\{D_i\}$ U $S_D$, where $D_i \notin S_D$

**Example 2:** Let $R$ be the following binary relation with 8 objects $\{O_1, O_2, O_3, O_4, O_5, O_6, O_7, O_8\}$ and three attributes $\{A,B,C\}$:

| | *A* | *B* | *C* |
|---|---|---|---|
| $O_1$ | 1 | 1 | 1 |
| $O_2$ | 1 | 1 | 0 |
| $O_3$ | 1 | 0 | 1 |
| $O_4$ | 1 | 0 | 0 |
| $O_5$ | 0 | 1 | 1 |
| $O_6$ | 0 | 1 | 0 |
| $O_7$ | 0 | 0 | 1 |
| $O_8$ | 0 | 0 | 0 |

$O_7$ is equivalent to $\{O_1, O_3 O_5\}$, the reason is that the concept containing $O_7$ is:
$RC = \{O_1,O_3, O_5, O_7\}$x{C; and inversely the concept containing $\{O_1,O_3,O_5\}$ is also $RC$.

$RC$ may be obtained by used the *Galois* Connection. This means that object $O_7$ can be removed without modifying the initial knowledge database $\{A,B,C,AB, AC,BC,ABC\}$ without any dependency between $A$, $B$ and $C$. By the same way, we can remove, $O_6$ and $O_4$. The reduced database $RD$ is the following:

| | A | B | C |
|---|---|---|---|
| $O_1$ | 1 | 1 | 1 |
| $O_2$ | 1 | 1 | 0 |
| $O_3$ | 1 | 0 | 1 |
| $O_5$ | 0 | 1 | 1 |

In that case, we can notice that in $RD$, we do not add or remove any new dependency, relatively to $R$.

This result is easy to generalize: because if for n attributes, we have $2^n$ objects, which means that we do not have any constraint between the n attributes, then we can replace $2^n$ objects (exponential function of $n$) by only $n+1$ objects (linear function of $n$), without changing the knowledge database. Unfortunately, the calculus of the reduction rate is not easy when the initial context contains functional dependencies between some of its attributes.

**The reduction algorithm:** The main motivation of our algorithm is to provide a data reduction technique that has the ability to carry out an exact reduction for application such as automatic reasoning, but also sufficiently loose and tight enough to identify non-exact match and to ensure that a true duplicates are detected in case of documents. The algorithm uses a threshold to identify the equivalence between document/object and a subset of documents/objects. A 100% threshold is used when the algorithm is used for exact data reduction, while less than 100% is used in the case of the non-exact but highly similar data reduction (documents and information retrieval systems). The threshold is used to identify the degree of duplication. The algorithm can be described as follow:

For each object $x$ in the domain calculates the overall degree (the degree of an object is the number of present properties or index terms). Construct a binary tree of the objects using the object degree as a key.

For each object $x$ in the domain of the remaining context, do the following steps:

* Find the set of all objects $S_x$ except $x$, with degree greater than or equal to the degree of $x$.
* Find the set of all objects Sx1 from $S_x$ that share all the properties of $x$.
* If $S_{x1}$ is not empty, check if object $x$ is up to a threshold included in the set of objects sharing the same properties as $S_{x1}$. In positive case, object $x$ is removed from the context (a collection of objects).

**Data reduction in automatic reasoning:** Expert systems generally use a knowledge database for making decision. Starting from initial facts, an expert system extracts additional facts using an inference engine. Expert systems are based on automatic forward, backward or mixed reasoning. We propose, another method based on reasoning by examples. We replace each association rule by a binary context, then, using relational join operator, we can obtain a total relation reflecting all possible cases, which are equivalent to the initial set of production rules. After this step, we apply reduction on the final table to minimize the number of objects. Then, starting from initial facts $A$, closure($A$)-$A$ is the total set of additional facts. The advantage of this method is that it is much faster than the other methods. But the problem is the size of the database, which may increase. We already know that join operator preserves functional dependencies and data[20]. Reduction operator does not preserve data, but it preserves knowledge.

**Example 3:** Suppose that the knowledge database is composed of two rules: $A \backslash B$ and $C \backslash D$. Then as a first step, we create two tables $T_1$ and $T_2$, where $T_1$ is composed of the minimal number of objects equivalent to $A \backslash B$ and $T_2$ is composed of the minimal number of objects equivalent to C\D.

| $T_1$ | *A* | *B* |
|---|---|---|
| | 0 | 0 |
| | 0 | 1 |
| | 1 | 1 |

| $T_2$ | $C$ | $D$ |
|---|---|---|
| | 0 | 0 |
| | 0 | 1 |
| | 1 | 1 |

$T_1 \infty T_2$ is the natural join of $T_1$ and $T_2$.

| | A | B | C | D |
|---|---|---|---|---|
| $O_1$ | 0 | 1 | 0 | 1 |
| $O_2$ | 0 | 1 | 0 | 0 |
| $O_3$ | 0 | 1 | 1 | 1 |
| $O_4$ | 0 | 0 | 0 | 1 |
| $O_5$ | 0 | 0 | 0 | 0 |
| $O_6$ | 0 | 0 | 1 | 1 |
| $O_7$ | 1 | 1 | 0 | 1 |
| $O_8$ | 1 | 1 | 0 | 0 |
| $O_9$ | 1 | 1 | 1 | 1 |

Using the reduction algorithm, we can remove objects $O_5$, $O_1$, $O_2$ and $O_4$, to only keep the following base of only 5 examples equivalent to the knowledge database of two rules.

| | $A$ | $B$ | $C$ | $D$ |
|---|---|---|---|---|
| $O_3$ | 0 | 1 | 1 | 1 |
| $O_6$ | 0 | 0 | 1 | 1 |
| $O_7$ | 1 | 1 | 0 | 1 |
| $O_8$ | 1 | 1 | 0 | 0 |
| $O_9$ | 1 | 1 | 1 | 1 |

Using the definition of closure of Section 2, we find that:

closure($\{A\}$) = $\{A,B\}$, closure($\{B\}$) = $\{B\}$, ($B \setminus$),
closure($\{A,B\}$) = $\{A,B\}$, ($A \setminus B$),
closure($\{A,C\}$) = $\{A,B,C,D\}$, ($A$ & $B \setminus C$ & $D$) ,
closure ($\{C,D\}$) = $\{C,D\}$, ($C$ & $D \setminus$), closure($\{D\}$) = $\{D\}$, (D $\setminus$),
closure ($\{A,D\}$) = $\{A,B,D\}$, (A & $D \setminus B$),
closure($\{B,C\}$) = $\{B,C,D\}$, ( $B$ & $C \setminus D$)
closure ($\{A,B,C\}$) = $\{A,B,C,D\}$, ($A$ & $B$ & $C \setminus D$),
closure($\{A,B,C,D\}$) = $\{A,B,C,D\}$ ($A$ & $B$ & C & $D \setminus$).

We can notice that we obtain all facts that might be concluded by any inference engine when it is directly applied on the knowledge database. This method has been applied on several real life examples. It always gives accurate results. Generalization to negation seems to be straightforward. For example, applied on a knowledge database of 11 rules and 16 attributes, we first obtain 90 objects. After reduction, we kept only 19 objects. Which means that the number of objects is less than the number of production rules.

**Data reduction in information retrieval:** Data reduction in information retrieval can be applied in two levels. First the duplicate document reduction can be applied to the documents collection in order to remove duplicate documents, thus saving storage space, improving the inverted index file and enhance the system precision. Second, the reduction algorithm could be applied on the collection of retrieved documents to remove redundant from the retrieved set, thus a saves user time and increases the system performance.

* Two experiments are conducted to investigate the efficiency of the reduction algorithm:
* The data reduction method is applied on the collection of documents.
* The data reduction method is applied on the set of retrieved documents in response to user queries.

## EXPERIMENTAL RESULTS

**Duplicate document reduction in a collection of documents:** In this experiment we investigate the performance of our duplicate document reduction method in reducing the size of the documents collection. We used two documents collections, as shown in Table 1. Each collection was carefully chosen to test particular issues involved with duplicate detection. The first collection contains 10,322 documents from the *Gulf Times* English newspaper. A subset of known duplicate document is inserted into the collection in order to analyze the performance of our algorithm in finding inserted duplicates. This collection is used to test the performance of our reduction algorithm on English documents and the ability to find duplicate inserted documents in the collection.

The second collection contains 10,340 documents from *Al-Raya* Arabic newspaper. This set was used to roughly mirror the English collection in terms of the number of documents and subjects but to contain Arabic documents. A subset of known duplicate document is also inserted to this collection to analyze the performance of our algorithm in finding inserted Arabic duplicates. This collection is used to test the performance of our data reduction algorithm on Arabic documents.

Table 1: Experimental collections

| Collection Name | Collection Size | Number of Documents |
|---|---|---|
| Gulf Times | 30 MB | 10,322 |
| Al-Raya | 36 MB | 10,340 |

Note that, there is no available information about the number of duplicate document in each collection, which make it too difficult to get any type of quantitative measure on the whole collection of our algorithm performance. This is not likely to change in the near future. As the document collections grow, the likelihood of judgments of duplicates being made is small; therefore our evaluation of the algorithm is based on the ability to find inserted subset of duplicate and the resulted size of the document collection after the data reduction algorithm is used.

To be able to get more performance measures, the insertion of different pre-known duplicate document subsets is repeated within the two collections before the algorithm is tested. These duplicate sets vary in number

of documents, length of the document, subjects and the degree of duplication (not exact repetition of the documents, or documents with minor differences such as author name and affiliation etc).

The first step in this experiment is to represent each document in the collection by a set of properties or index terms. We used automatic indexing with inverse frequency term waiting approach proposed by Salton[21] to identify the properties of each document before running the algorithm. As a result, each document is represented by a vector of zeros and ones where 1 in the location i means that the term $T_i$ is relevant to represent the document contents.

Table 2 and 3, give the result of our algorithm performance in identifying duplicate document in the two collections. As we can see from the tables, our algorithm manages to get an average of 96.5% detection percentage in the English collection and an average of 94% detection percentage in the Arabic Collection. The algorithm performance decreases slightly with Arabic document collection. This decrease in the performance is due to the heavily use of synonym in the Arabic writing which makes two almost identical documents having lower similarity measures because of the different set of vocabulary used. Table 4, gives the number of documents retained as unique documents after the run of the reduction algorithm. As a result, the inverted index file size and storage space is reduced. Thus saving storage space and give smaller and more precise index file that can be searched more efficiently.

Table 2: Algorithm found ratio for the English collection

| Duplicate document set number | Found Ratio |
|---|---|
| Set 1 | 100% |
| Set 2 | 90% |
| Set 3 | 94% |
| Set 4 | 100% |
| Set 5 | 100% |
| Set 6 | 93% |
| Set 7 | 95% |
| Set 8 | 100% |
| Average | 96.5% |

Table 3: Algorithm found ratio for the Arabic collection

| Duplicate document set number | Found Ratio |
|---|---|
| Set 1 | 87% |
| Set 2 | 93% |
| Set 3 | 100% |
| Set 4 | 97% |
| Set 5 | 100% |
| Set 6 | 85% |
| Set 7 | 96% |
| Set 8 | 94% |
| Average | 94% |

Table 4: Number of unique documents in the Arabic and English collections

| Collection | Original number of documents | Unique documents found in the collection |
|---|---|---|
| Gulf Times English Collection | 10,322 | 7019 |
| Al Raya Arabic Collection | 10,340 | 7238 |

**Duplicate document reduction in information retrieval systems:** In this experiment we investigate the effect of duplicate document reduction on the performance of the information retrieval systems. Our information retrieval system is based on the vector space model, where each document is represented by a vector in the n-dimension space. The first step in this experiment is to index the document collection using the automatic indexing with inverse frequency term weighting approach[22]. The indexing process was carried out as follows:

* Eliminate common function words from the document texts by consulting a stop list containing a list of high frequency function words.
* Compute the term frequency $tf_{ij}$ for all remaining terms $T_j$ in each document $D_i$.
* Assign to each document $D_i$ all terms $T_j$, such as $T_{ij}$ greater than a threshold.

The retrieval process is carried out by calculating the similarity between the document vectors and the query vector, documents with similarity greater than a threshold are retrieved. The second step is to collect a set of query to be used for system evaluation. A set of students in their senior year at the University of Qatar, who are regular newspaper readers, supplied us with thirty Arabic queries. This set of queries is used to carry out the following experiments:

**Arabic monolingual information retrieval:** In this experiment the set of queries is supplied to the system and documents with similarity greater than a threshold are returned to the user in response to his query.

**Arabic monolingual information retrieval with duplicate document reduction mechanism:** In this experiment the set of queries is supplied to the system and the documents with similarity greater than a threshold are identified. Second, our reduction algorithm is run on the set of identified relevant documents. The set of resulted documents is then returned to the user in response to his queries.

**English monolingual information retrieval:** In this experiment an expert translator translates the set of Arabic queries to English before the queries are supplied to the system with the English documents collection to identify the relevant documents.

**English monolingual information retrieval with duplicate document reduction mechanism:** This experiment is the same as experiment two except that the queries are in English and matched with the documents in the English collection.

We asked the students of the computer science department at the University of Qatar (340 students) to record how many citations they look in a retrieval task using different Internet search engines. The students ranged between twenty to thirty first retrieved

documents on the first search. If the retrieved documents in the first twenty citations were not satisfactory the students refined their queries instead of looking for more citations. Using the outcome of this experiment we focused our evaluation of retrieval approaches described here on the first fifty retrieved documents. The total number of documents retrieved will be at most fifty according to the terms of our experiment. In addition, the huge number of documents in each collection make it infeasible to carry out the relevance judgment process.

Our system evaluation is based on the precision measure. The duplicate retrieved documents are considered irrelevant in the sense of no addition information gain and wasting the user time. The query precision is calculated using the following formula (Baeza-Yates and Ribeiro-Neto[22]):

The average precision of the system is calculated using the precision of the thirty different queries supplied to the system. The average precision of the two Arabic monolingual information retrieval experiments is given in Table 5.

Table 5: Average precision for the two Arabic monolingual retrieval experiments

| Experiment | Average Precision |
|---|---|
| Arabic Monolingual without duplicate document detection mechanism | 0.473 |
| Arabic Monolingual with duplicate document detection mechanism | 0.544 |

The average precision of the information retrieval system is increased by 15% with the use of the duplicate document reduction mechanism. As a result, it is clear that the use of data reduction approaches improve the quality of the Arabic information retrieval system and would provide the user with un-repeated relevant information in the top ranked set of documents.

Table 6 gives the average precision of the two English monolingual retrieval experiments. As shown in the table the retrieval system precision is enhanced with the removal of duplicate document. The precision measure increased by 16.5%.

Table 6: Average precision for the two English monolingual retrieval experiments

| Experiment | Average Precision |
|---|---|
| English Monolingual without duplicate document detection mechanism | 0.517 |
| English Monolingual with duplicate document detection mechanism | 0.602 |

## CONCLUSION

We have proposed a new data reduction algorithm using concept analysis which can be used as a filter in retrieval systems like search engines to eliminate redundant references to the similar documents. We have also studied the application of the algorithm in automatic reasoning which resulted in minimizing the number of stored facts without loosing of knowledge. Two experiments have been carried out in information retrieval systems where the first one consisted of evaluating the performance of the algorithm in the detection and removal of duplicate document from a collection of Arabic and English documents. The algorithm scored a high ratio in detecting and removing duplicate document in both languages. The second experiment consisted of studying the effect of using the algorithm as part of an information retrieval system for both Arabic and English documents. Our results showed a good increase in the retrieval system precision in addition to reducing the user time and increases his satisfaction.

## REFERENCES


1. Broder, A.Z., S.C Glassman, M.S. Manasse and G. Zweig, 1997. Syntactic clustering of the web. Sixth Intl. World Wide Web Conf.
2. Kulkarni, P., F. Douglis, J.D. LaVoie and J.M. Tracey, 2004. Redundancy elimination within large collections of files. USENIX Ann. Tech. Conf., General Track, pp: 59-72.
3. Mogul, J.C., Y.-M. Chan and T. Kelly, 2004. Design, implementation and evaluation of duplicate transfer detection in http. NSDI, pp: 43-56.
4. Shaivakumar, N. and H. Garica-Molina, 1998. Finding near-replicas of documents on the web. Proc. Workshop on Web Databases.
5. Yue, L., C.L. Tan, 2004. Information retrieval in document image databases. IEEE Trans. Knowledge and Data Engg., 16: 1398-1410.
6. Ganter, B. and Wille, 1999. Formal Concept Analysis. Springer-Verlag.
7. Mineau, G.W. and R. Godin, 1995. Automatic structuring of knowledge bases by conceptual clustering. IEEE Trans. Knowledge and Data Engg., 7: 824-829.
8. BenYahia, S., K. Arour, A. Slimani and A. Jaoua, 200. Discovery of compact rules in relational databases. Information J., 3: 497-511.
9. Jaoua, A. and S. Elloumi. Galois connection, formal concept and galois lattice in real binary relation. To appear in the Journal of Systems and Software.
10. Ben Yahia, S. and A. Jaoua, 2001. Discovering knowledge from fuzzy concept lattice. Kandel, A., Last, M. and Bunke, H, Eds., Data Mining and Computational Intelligence, Studies in Fuzziness and Soft Computing, Vol. 68, Chap. 7, Physica Verlag, Heidelberg, pp: 167-190.
11. Brin, S., J. Davis and H. Carcia-Molina, 1995. Copy detection mechanisms for digital documents. Proc. SIGMOD'95.

12. Heintze, N., 1996. Scalable document fingerprinting. Proc. 2nd USENIX Workshop on Electronic Commerce.
13. Buckley, C., C. Cardie, S. Mardis, M. Mitra, D. Pierce, K. Wagstaff and J. Walz, 2000. The Smart/Empire TIPSTER IR System. TIPSTER Phase III Proceeding, Morgan Kaufmann.
14. Chowdhury, A. and O. Frieder *et al*., 2002. Collection statistics for fast duplicate document detection. ACM Trans. Information Systems, 20: 171-191.
15. Sanderson, M., 1997. Duplicate detection in the reuters collection. Technical Report (TR-1997-5) of the Department of Computing Science at the University of Glasgow, Glasgow G128QQ, UK.
16. Salton, G., C.S. Yang and A. Wong, 1975. A vector-space model for information retrieval. Commun. the ACM, 18.
17. Mcleod, R., 2000. Management Information Systems: A Study of Computer Based Information System. 7th Edn.
18. Scotti, R. and C. Lilly, 1999. George Washington University Declassification Productivity Research Center. http://dprc.seas.gwu.edu.
19. Jaoua, A, K. Bsaies and W. Consmtini, 1999. May reasoning be reduced to an information retrieval problem. Relational Methods in Computer Science, Quebec, Canada.
20. ElMasri, R. and S.B. Navathe, 2000. Fundamentals of Database Systems. Third Edn. Addison-Wesley.
21. Salton, G. and M. McGill, 1983. Introduction to Modern Information Retrieval. McGraw-Hill, New York.
22. Baeza-Yates, R.A., R. Baeza-Yates and Berthier Ribeiro-Neto, 1999. Modern Information Retrieval. Addison-Wesley Longman Publishing Co., Inc., Boston, MA.