Dynamic Clustering for Information Retrieval from Big Data Depending on Compressed Files

Dr.Alaa Kadhim F.
Computer Sciences Department,
University of technology/ Baghdad,
Iraq

Prof. Dr. Ghassan H. Abdul Majeed
Ministry of higher education/
Baghdad, Iraq

Rasha Subhi Ali
Computer Sciences Department,
University of technology/ Baghdad,
Iraq

***Abstract*—The rapid growth in the database data led to origination a large amount of data. So, it is still a big problem to access this data for answering user queries. In this paper a novel approach for aggregating the required data was proposed, this approach called dynamic clustering. Also, several retrieval methods were used for retrieving purposes. The dynamic clustering method is built clusters according to the user entries (queries). It has been applied to different compressed database files in different size and using different queries. The compressed database file is resulted from applying ICM (Ideal Compression Method) and best compressed algorithm (improved k-mean, k-mean with medium probability and k-mean with maximum gain ratio). The retrieval methods applied to original database file, compressed file and the cluster that result from implementing dynamic clustering algorithm and the results was compared.**

Keywords—dynamic clustering; data retrieval methods; compression algorithm; ICM system; improved k-means algorithm and modified improved k-means algorithms

I. Introduction

Big Data interests with large-volume, complex, growing data sets in multiple and autonomous sources. Data storage and Data collection have become more complex. Big Data is now expanding quickly in all science and engineering domains, including physical, biological and biomedical sciences, etc. The most essential challenge for Big Data applications is to explore the large amount of data and extract useful information or knowledge for future actions [1]. The goals of data compression is the task of providing space on the hard drive, and reduce the use of bandwidth in the transmission network and transfer files quickly. Data compression purposes are to reduce the number of bits used to store or transmit data. Data Compression methods are divided into two types 1) lossless compression method and 2) lossy compression method [2].In this paper we used Lossless data compression techniques. In lossless data compression, the integration of data is preserved without loss any information. In this paper the ICM system results are used to build the dynamic clusters.

The Data Mining is defined as an extraction of hidden information from large databases. It is a powerful advanced technology. It has great possibility helps the Libraries and information centers to focus on the most important information in their data warehouse. There are several techniques for data mining these are: 1) classification 2) clustering 3) prediction (regression) 4) decision trees 5) sequential patterns and 6) association rules [3]. In this paper the clustering technique was used to solve the problem of accessing big data. The ICM system was used to specify the best compression algorithm and the results of the compression algorithm is used to build the dynamic clusters. Clustering technique is one of the most important data mining techniques. The aim of cluster analysis is to divide the data (objects, instances, items) into groups (clusters) so that items belonging to the same group are more similar than items belonging to distinct groups [2]. Also, the retrieval methods were used to answering user queries. The most frequently used operations on transactional databases is the data retrieval operation. Data retrieval means obtaining data from a database management system In order to retrieve the desired data the user specifies a set of criteria by a query. Then the Database Management System (DBMS) selects the required data from the database. There are several searching strategies for data retrieval these are keyword, Boolean operators, truncation, phrase searching, and search limiting, and nesting. All search strategies are based on comparison between the query and the stored documents [4]. The retrieved data may be stored in a file, printed, or viewed on the screen. In traditional database management systems, information retrieval is often performed using keywords contained within fields of each record [5]. So, for faster retrieving the compression methods were used to compress the database files and the dynamic clustering method was used to build clusters contain information about the required query data, so the retrieving data become much faster than the retrieving from original and compressed files. Numerous studies and tools already can be found in the scientific literature.

The system proposed by S. Parthasarathy, V. Shakila [4] focuses mainly on a k_means algorithm that characterizes the features of the Big Data revolution, and proposed a Big Data processing model from the Data mining perspective. This provided the most relevant and most accurate social sensing feedback to better understand our society at real time with big data technologies. The system proposed by Hazem M. El-Bakry, Nikos E. Mastorakis, Michael E. Fafalios [6] focuses on using an efficient model for fast retrieving of specific information from big data. Fast neural networks are used to find the best matching between words in query and stored big data. The idea is to accelerate the searching operation in a big data. This is done by applying cross correlation between the given query and the big data in the frequency domain rather than the time domain. The research work proposed by QIAN WeiNing, GONG XueQing and ZHOU AoYing [7] describes using a hybrid-clustering algorithm to solve the problems of scanning the whole database, pre-specifying the uncertain parameter k and lacking high efficiency in treating arbitrary shape under very large data set environment. The proposed algorithm combines both distance and density strategies, handled any arbitrary shape clusters effectively. It makes full used of statistical information in mining to reduce the time complexity greatly while keeping good clustering quality.

This research is organized as follows. Section one shows the introduction, section two presents data compression, Section three explains major clustering techniques , section four explains major data retrieval methods, Section five shows the methodology of dynamic clustering method and data retrieval methods and system structure, Section six presents experiments and results, section six offers the conclusion and in section seven suggests a future works.

II. Compression Methods

Data compression purposes are to reduce the number of bits used to store or transmit data. Data Compression is basically defined as a technique to reduce the size of data by applying different methods that can either be Lossy or Lossless. Data compression is popular for two reasons: (1) People like to collect data and hate to throw anything away. (2) People hate to wait a long time for data transfers. Compression process may be useful if one wants to save the storage space. For example, if one wants to store a 4MB file, it may be best to compress it to a smaller size to provide the storage space. Also compressed files are much more easily exchanged over the internet since they upload and download much faster [2]. The main goal of data compression is to reduce the redundancy in warehouse or communicated data. Lossless Compression is used when the original data from a source are so important retrieved without lose any details. The main purpose of this compression technique is to compress the file by decreasing the information in such a way that there is a no loss when decompress any file back into the original file. Example of lossless data compression technique is text compression [8]. A lossy data compression method is one where the retrieved data after decompression may not be exactly same as the original data, but is "close enough" to be useful in particular purpose [2]. In this paper the clustering technique (Improved K-means, K-means With Medium Probability and K-means With Maximum Gain Ratio) algorithms were used as lossless compression algorithm and the results have been used to build the dynamic clusters.

III. Data Clustering

Data mining, the extraction of hidden predictive information from large databases, is a powerful technology with great possibility to help companies concentrate on the most important information in their data warehouses. Data mining tools can answer business queries that traditionally consumed too long time to resolve [9]. Cluster analysis is a mechanism for multivariate analysis that assigns items to create groups based on a calculation of the degree of association between items and groups. There are two main types of cluster analysis methods these are the nonhierarchical, which divide a dataset of N items into M clusters, and the hierarchical, which output nested dataset in which pairs of items or clusters are successively linked. In the information retrieval (IR) field, cluster analysis has been used to create groups of documents with the goal of benefiting the efficiency and effectiveness of retrieval [10]. Clustering is a data mining technique that makes useful cluster of objects which have similar characteristics using automatic technique [3]. In this paper the (Improved K-means, K-means With Medium Probability and K-means With Maximum Gain Ratio) algorithms were used to build clusters for the inputted database and dynamic clustering method was used to build clusters according to the used entries (queries). The benefit of using (Improved K-means, K-means With Medium Probability and K-means With Maximum Gain Ratio) algorithms as compression methods is to reduce the file size then the required answering time for user queries was decreased. As well, the advantage of building dynamic clusters for entries of user query led to speed up the data retrieval operation (speed up answering for user query).

K-means is one of the most commonly used clustering techniques due to its simplicity and speed. It partitions the data into k clusters by assigning each object to its closest cluster centroid (the mean value of the variables for all objects in that particular cluster) based on the distance measure used. The basic algorithm for k-means works as follows:

```
Algorithm 1 k-mean [11]
Input: C: the number of cluster and D: A data set containing m objects.
Output: A set of C cluster.
Begin:
  Choose m objects randomly from dataset as the initial cluster centers;
    Until there are no changes in the mean values
        Use the estimated means to classify m objects into k clusters based on similarity measured
        For i=1 to k
            Calculate mean value of the objects for each cluster i and make replacing old mean with new mean
        End_for
    End_until
End
```

Usually, the K-means algorithm criterion function depends on the square error criterion, which can be defined in the following equasion:

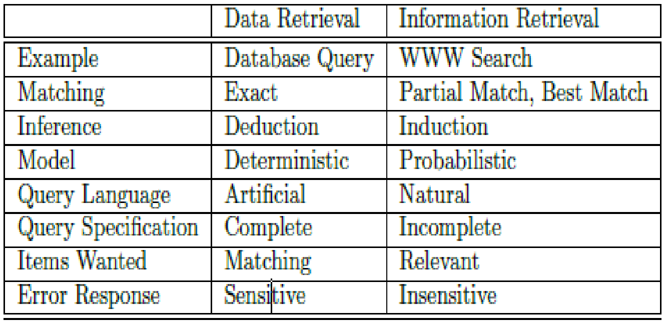
$$\mathbf{E} = \sum_{j=1}^{k} \sum_{\substack{i=1 \\ x_i \in c_j}}^{n} \left\| x_{i-m_j} \right\|^2 \tag{1}$$

In which, E is the total square error of all the objects in the data cluster, xi is the vector of the i-th element of the dataset, mi is the mean value of cluster Ci (x and m are both multi-dimensional). K-means is the most important clustering technique that has been used widely in the field of IR. It was grouped data objects into k clusters [12]

IV. Data Retrieval

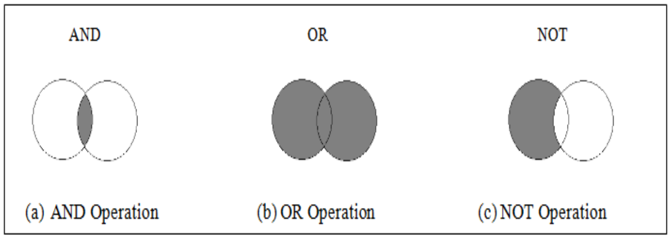
Databases are electronic collections of information, the databases were used to retrieve items in a catalog or a periodical database. Each item in a database is a record. Each record consists of a set of fields [13].A database-management system (DBMS) is a collection of correlated data and a set of programs to access those data. Usually a collection of data referred to it as the database, contains relevant information to an enterprise. The essential goal of a DBMS is to provide a way to store and retrieve database information that is both convenient and efficient [14].Information Retrieval is the technique of presentation, storage, organization of and access to information items. The representation and organization of information should be in such ways that the users can access information to meet their information need [15]. Data retrieval means obtaining data from a database management system. The retrieved data may be stored in a file, printed, or viewed on the screen. The retrieval process has been begun with the user entering a query. The query entered by the user can be a one word or it can be a sentence [16]. Information retrieval (IR) is finding items (usually documents) of an unstructured nature (usually text) that meets an information need from within large collections (usually stored on computers) [15]. The difference between information retrieval and data retrieval is summarized in the following table:

TABLE I. THE DIFFERENCE BETWEEN IR AND DATA RETRIEVAL [15]



	Data Retrieval	Information Retrieval
Example	Database Query	WWW Search
Matching	Exact	Partial Match, Best Match
Inference	Deduction	Induction
Model	Deterministic	Probabilistic
Query Language	Artificial	Natural
Query Specification	Complete	Incomplete
Items Wanted	Matching	Relevant
Error Response	Sensitive	Insensitive

Information retrieval (IR) systems uses a simpler data model than database systems, it was information organized as a collection of documents (documents are unstructured data). While Database systems deal with structured data, with schemas that define the data organization [14]. Searching strategies include: keyword and subject searching, Boolean operators, truncation, phrase searching, search limiting, and nesting [4].Boolean searching is a method based on logic. Most online databases and internet search engines based on Boolean searches. The Boolean operators AND, OR, NOT (or AND NOT). Using AND narrows your search. It retrieves records that contain both of the search items or keywords that you specify. The more items (terms or keywords) connected with (AND) the fewer search results will be found. Using OR expand your search. It retrieves records that contain either of the search items (terms) or keywords that you specify, but not necessarily both. The more items (terms) connected with OR, the more search results will be found. Using NOT narrows the search. It retrieves records that do not contain a search item (term) in your search. NOT was used to exclude a term from your search and to find fewer results [17]. The bellows figure shows these operations.



(a) AND Operation (b) OR Operation (c) NOT Operation

Fig. 1. Boolean Operation

The index is a logical view where documents or data in a collection are represented through a set of index terms or keywords, i.e., any word that appears in the document text or field in the database[18]. An index for a file in a database system works in much the same way as the index in the textbook [14]. Keyword Searching best used method for searching new terms (items), special words, jargon or slang. Phrase searching is a way to retrieve records containing specific phrases. A phrase search will locate only records containing the specified (inputted) words [4]. Keyword query is easy and flexible because it doesn’t require from the database user to know details about the database schema. The goal of information retrieval is to identify documents which best match user needs. While the goal of data retrieval is to identify table records which best match user needs [5].

V. PROPOSAL DYNAMIC CLUSTERING SYSTEM

The proposed system is dependent on a dynamic clustering algorithm. Clustering is a collection of items that have high similarity measures in the same cluster and dissimilar to the items belonging to other cluster. The traditional k-means algorithm problems it was needed to specify the center points and the number of the clusters by the user, it was also needed to calculate similarity measures and this will take a long time and it was applied on numerical data only. The (Improved K-means, K-means With Medium Probability and K-means With Maximum Gain Ratio) algorithms were used to compress the database and were built clusters according to the specified centers. The centers have been specified in dynamic way according to the best selected compression algorithm. The selection was conducted by applying the ICM algorithm. The ICM algorithm is worked by analyzing the database, then depending on the extracted features of the database the ICM would be selected best compression algorithm (Improved K-means, K-means With Medium Probability and K-means With Maximum Gain Ratio). We proposed a new algorithm called a dynamic clustering algorithm. This algorithm builds a cluster depending on user entries. The final result of ICM was represented by the compressed file that was resulted from selecting and applying the best compression algorithm. This file was used to build dynamic clusters based on user query (user entries). The dynamic clusters contain data that are relevant to the user queries only. Example voters' information about voters' lives in Baghdad, year nascent like 1990 and name polling station. The results of dynamic clusters have advantages in data retrieval system, it was much faster in retrieving data than retrieving from original database or compressed file because of the searching was worked in less data than the data contained in original database or in compressed data. The retrieval of data from the large database forms a problem because it was taken much time. So the compression and building dynamic clusters solved this problem, it was taking lesser time in retrieving the required data. Several retrieval methods were used in retrieving data, these methods are: 1) indexing, 2) keyword searching and 3) applying Boolean operations. These retrieving methods have been applied on original database, compressed file and dynamic clustering file and the results were compared. Dynamic clustering includes of making clusters based on specified user query data (entries of the query). The clusters are made from the compressed database file and not from the original database file. There exists a historical file which consists of all the previous user queries for the built clusters. Therefore the stored clusters and its stored query data were not needed to make it again. When the user enters query the system do searches in the historical file, if it was found the query in the historical file then it is returned the answer for the user query directly, else it was making a new cluster for the entering query and then the answer returned for the user query.

The proposed system consists of several phases. These steps consist of:

1) Input the original database file
2) Apply the ICM algorithm
3) Compress the database file with the best compression method
4) Returning compressed file
5) Input the user query to the dynamic clustering algorithm
6) Analyzer: searching the user query if it was existed in the historical file or not and
7) Return the results

These steps can be explained in the following phases:

Phase 1: This phase includes selecting the database file to be inputted to ICM. The inputted database can be contained any type of data (numerical data, text data, .etc.).

Phase 2: This phase compromises the selection the optimal compression method. The ICM algorithm was used for this purpose. The ICM algorithm depends on the extracted features of the database by analyzing the inputted database. Also, this algorithm depends on several conditions used to specify the best compression method (improved k-mean, k-mean with medium probability and k-mean with maximum gain ratio). Each one of these algorithms depended on specific calculations to specify the centers of the clusters.

Phase 3: The ideal compression method was selected in the previous phase. In this phase the best compression method can be applied to the entering database.

Phase 4: The results of the compression algorithm are recorded in this phase. These results have been represented by the compressed file contains database data in clusters form.

Phase 5: At this phase the users entering the query data to the dynamic clustering algorithm.

Phase 6: This phase includes analyzing the user query by searching in historical file if the user query exists or not.

Phase 7: this is the final phase consists of displaying the query answers or making new clusters and then displaying query answers. If the query not existed in historical file, then making new clusters and saving the user query in historical file then return the query answer to the user. The proposed system can be explained in the following architecture.

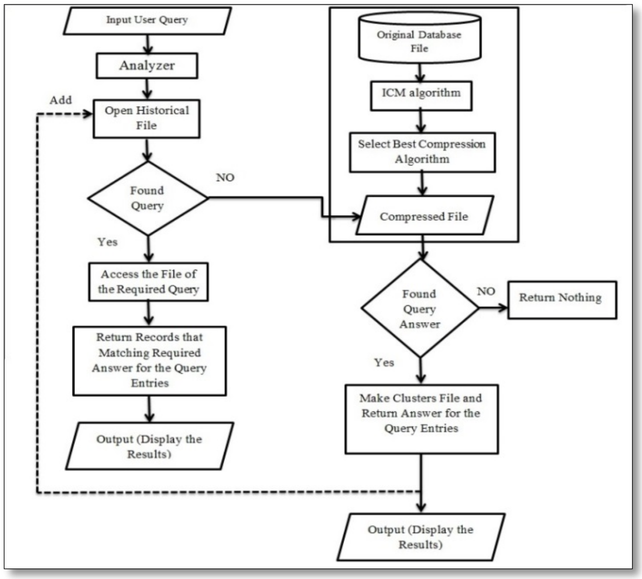

Input User Query
Analyzer
Add
Open Historical File
Found Query
NO
Yes
Access the File of the Required Query
Return Records that Matching Required Answer for the Query Entries
Output (Display the Results)
Original Database File
ICM algorithm
Select Best Compression Algorithm
Compressed File
Found Query Answer
NO
Return Nothing
Yes
Make Clusters File and Return Answer for the Query Entries
Output (Display the Results)

Fig. 2. Proposed System Architecture

The structure of the dynamic clustering system is considered by entering the users for queries data to the analyzer. The analyzer would be done searching in the historical file to specify if the entered queries exist or not. If the queries exist, the system would be returned the required data, else the system would be built new clusters for the required queries and after that the system would be returned the required data. The analyzer has two situations, these are classified as the existed of the user query and not existed of the user query in the historical file. Results can be returned directly or indirectly. The results returned directly if the analyzer in the case of found the query in the historical file. The results returned indirectly if the analyzer in the case of not found the query data in the historical file, this is because of the system was built new clusters and then return the results. The dynamic clustering steps can be explained in the following algorithm:

Algorithm 2: Dynamic clustering algorithm

```
Input: user query.
Output: answers return the required records that match the user query.
Begin:
Open the historical file to check if the query exists or not exists in the historical file.
If the query exists in the historical file then
    Fetch the path of the file that contains the data of the entered query and then open this file let it x.
    While not end of x do
        Search about the required query using (keyword strategy; indexing strategy or phrase and Boolean operation strategy).
        Return all the records that match the required query data.
    End while.
Else If the query not exists in the historical file then
    While not end of the compressed file
        Search about the required query using (keyword strategy; indexing strategy or phrase and Boolean operation strategy).
        Open the compressed file and match the query data with data in the compressed file.
        Open new file for saving clusters that was extracted from matching the user query.
        Save user query in historical file.
    End while
End If
Display the results to the user.
End.
```

The retrieval from the file that was resulted from applying dynamic clustering algorithm much faster than retrieval from original database or compressed database file. In this research the string matching method was used. It was too fast method. The comparison between the words has been applied to find the matching words that match the required entries of user queries. The character comparison is a fast searching method to search the required text or records. The entity search includes of entry from one or more columns. Also, the indexing method was used to search about specific record. In addition to the indexing and string matching search methods, the Boolean operation, keyword searching, SQL query and phrase searching strategies have been used to search about the required queries in clustered and non-clustered data structure. The Boolean (logical) operation was used to search about more than one item, each item located in different columns. The indexing strategy was used to search about distinctive identifier (special ID). Keyword strategy was used to search about records that containing the required keywords (e.g. voters' how have year nascent =1990) and SQL was used to search in the original database file. The structure of dynamic clustering method can be explained in the below figure.

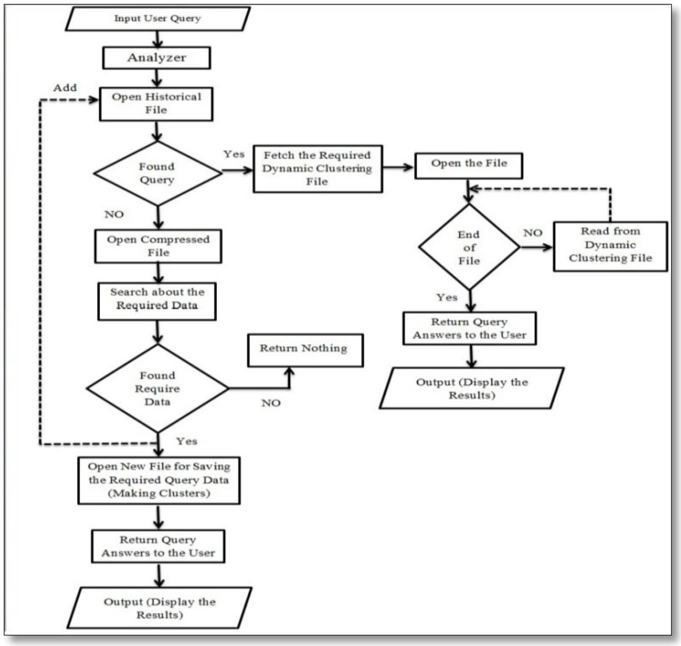

Input User Query
Analyzer
Add
Open Historical File
Found Query
Yes
Fetch the Required Dynamic Clustering File
Open the File
NO
Open Compressed File
End of File
NO
Read from Dynamic Clustering File
Search about the Required Data
Yes
Return Query Answers to the User
Return Nothing
Found Require Data
NO
Output (Display the Results)
Yes
Open New File for Saving the Required Query Data (Making Clusters)
Return Query Answers to the User
Output (Display the Results)

Fig. 3. The structure of dynamic clustering method

VI. Results and Discussion

This section discusses the results that obtained from the new proposed method (dynamic clustering). The dynamic clustering method was applied on different data sets for different data types, and was conducted different experiments to specify the performance of the proposed method.

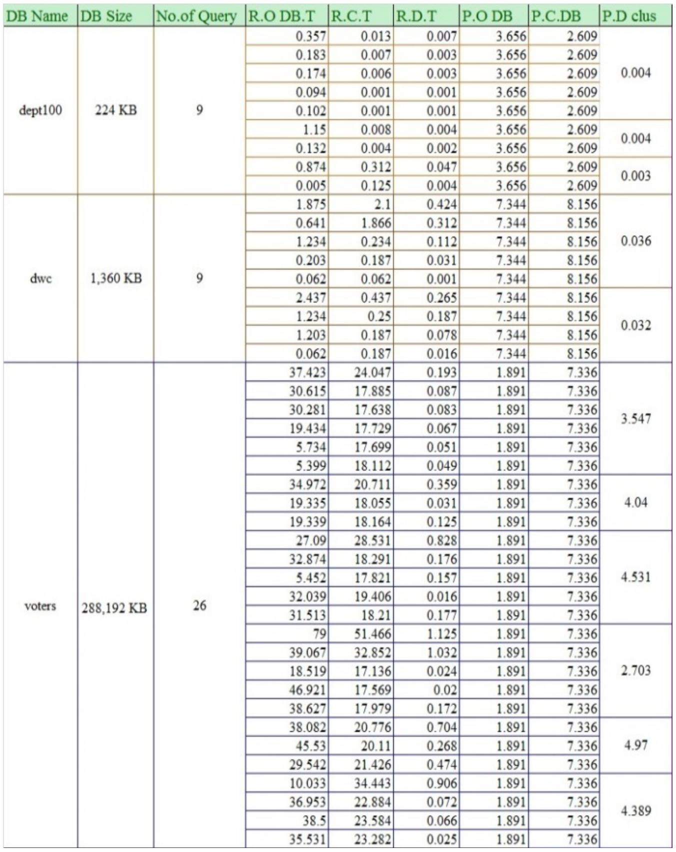
The experiments have been conducted on different databases and the results are compared based on the consumed time in retrieving from original database file, compressed database file and dynamic clustering file. The proposed method results showed in the following tables. The time was measured in seconds (e.g. 40.11seconds=40110 milliseconds and 0.016 seconds=16 milliseconds). Tables [2 and 3] show the detailed results for the tested databases such as:

* DB Name (database name),
* DB Size (database size),
* NO of Query (number of query),
* R.O DB.T (retrieval time from original database file),
* R.C.T (retrieval time from the compressed file),
* R.D.T (retrieval time from building dynamic clustering file),
* P.O DB (preprocessing operation time for original database file),
* P.C.DB (preprocessing operation time for compressed database),
* P.D clus (preprocessing operation time for dynamic clustering),
* T.T.R.O DB (total time for retrieving from original database file),
* T.T.R.C DB (total time for retrieving from the compressed database file) and

* T.T.R.D clus (total time for retrieving from dynamic clustering resulted file.

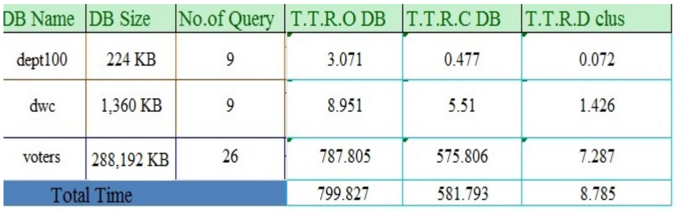
From table (2) we would notice that the consumed time to answering about the user query in the state of querying from the dynamic clustering file is too much faster than answering in the situation of querying from the original or compressed database file. This is because of the dynamic clustering file contains the data that had related with only user query. Optimization proposal is advances by reducing the required vocabulary check about the user query.

TABLE II. PROPOSED SYSTEM ANSWERING TIME RESULTS

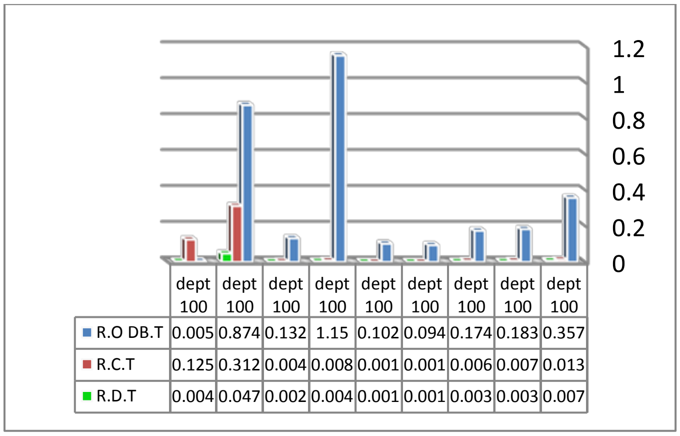


DB Name	DB Size	No.of Query	R.O DB.T	R.C.T	R.D.T	P.O DB	P.C.DB	P.D clus
dept100	224 KB	9	0.357	0.013	0.007	3.656	2.609	0.004
			0.183	0.007	0.003	3.656	2.609	
			0.174	0.006	0.003	3.656	2.609	
			0.094	0.001	0.001	3.656	2.609	
			0.102	0.001	0.001	3.656	2.609	
			1.15	0.008	0.004	3.656	2.609	0.004
			0.132	0.004	0.002	3.656	2.609	
			0.874	0.312	0.047	3.656	2.609	0.003
			0.005	0.125	0.004	3.656	2.609	
dwc	1,360 KB	9	1.875	2.1	0.424	7.344	8.156	0.036
			0.641	1.866	0.312	7.344	8.156	
			1.234	0.234	0.112	7.344	8.156	
			0.203	0.187	0.031	7.344	8.156	
			0.062	0.062	0.001	7.344	8.156	
			2.437	0.437	0.265	7.344	8.156	0.032
			1.234	0.25	0.187	7.344	8.156	
			1.203	0.187	0.078	7.344	8.156	
			0.062	0.187	0.016	7.344	8.156	
voters	288,192 KB	26	37.423	24.047	0.193	1.891	7.336	3.547
			30.615	17.885	0.087	1.891	7.336	
			30.281	17.638	0.083	1.891	7.336	
			19.434	17.729	0.067	1.891	7.336	
			5.734	17.699	0.051	1.891	7.336	
			5.399	18.112	0.049	1.891	7.336	
			34.972	20.711	0.359	1.891	7.336	4.04
			19.335	18.055	0.031	1.891	7.336	
			19.339	18.164	0.125	1.891	7.336	
			27.09	28.531	0.828	1.891	7.336	4.531
			32.874	18.291	0.176	1.891	7.336	
			5.452	17.821	0.157	1.891	7.336	
			32.039	19.406	0.016	1.891	7.336	
			31.513	18.21	0.177	1.891	7.336	
			79	51.466	1.125	1.891	7.336	2.703
			39.067	32.852	1.032	1.891	7.336	
			18.519	17.136	0.024	1.891	7.336	
			46.921	17.569	0.02	1.891	7.336	
			38.627	17.979	0.172	1.891	7.336	
			38.082	20.776	0.704	1.891	7.336	4.97
			45.53	20.11	0.268	1.891	7.336	
			29.542	21.426	0.474	1.891	7.336	
			10.033	34.443	0.906	1.891	7.336	4.389
			36.953	22.884	0.072	1.891	7.336	
			38.5	23.584	0.066	1.891	7.336	
			35.531	23.282	0.025	1.891	7.336	

TABLE III. PROPOSED SYSTEM TOTAL TIME RESULTS

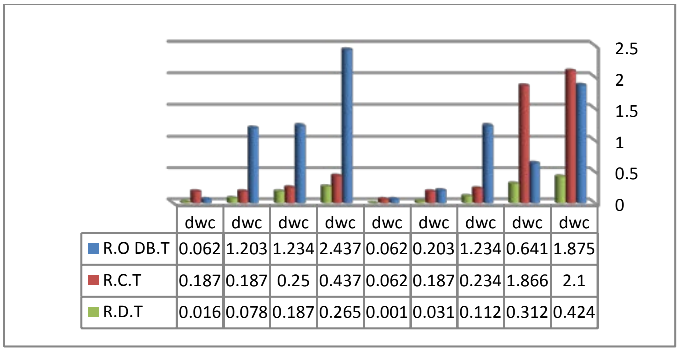


DB Name	DB Size	No.of Query	T.T.R.O DB	T.T.R.C DB	T.T.R.D clus
dept100	224 KB	9	3.071	0.477	0.072
dwc	1,360 KB	9	8.951	5.51	1.426
voters	288,192 KB	26	787.805	575.806	7.287
Total Time			799.827	581.793	8.785



	dept 100	dept 100	dept 100	dept 100	dept 100	dept 100	dept 100	dept 100	dept 100
■ R.O DB.T	0.005	0.874	0.132	1.15	0.102	0.094	0.174	0.183	0.357
■ R.C.T	0.125	0.312	0.004	0.008	0.001	0.001	0.006	0.007	0.013
■ R.D.T	0.004	0.047	0.002	0.004	0.001	0.001	0.003	0.003	0.007

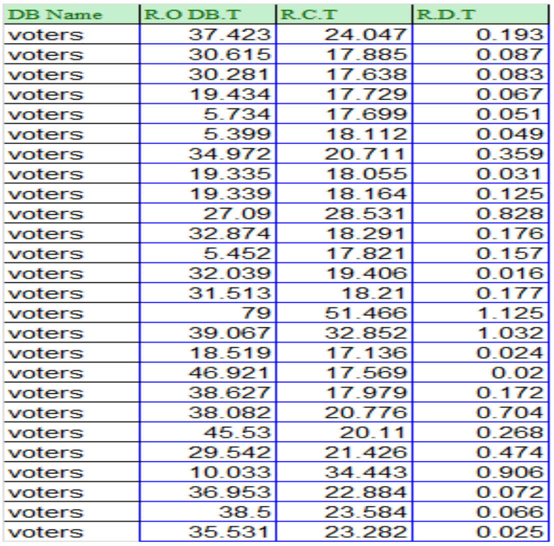
Fig. 4. Retrieval time for database dept100 of size 224 KB



	dwc	dwc	dwc	dwc	dwc	dwc	dwc	dwc	dwc
■ R.O DB.T	0.062	1.203	1.234	2.437	0.062	0.203	1.234	0.641	1.875
■ R.C.T	0.187	0.187	0.25	0.437	0.062	0.187	0.234	1.866	2.1
■ R.D.T	0.016	0.078	0.187	0.265	0.001	0.031	0.112	0.312	0.424

Fig. 5. Retrieval time for database DWC of size 1,360 KB

TABLE IV. RETRIEVAL TIME FOR DATABASE VOTERS OF SIZE 288,192 KB



DB Name	R.O DB.T	R.C.T	R.D.T
voters	37.423	24.047	0.193
voters	30.615	17.885	0.087
voters	30.281	17.638	0.083
voters	19.434	17.729	0.067
voters	5.734	17.699	0.051
voters	5.399	18.112	0.049
voters	34.972	20.711	0.359
voters	19.335	18.055	0.031
voters	19.339	18.164	0.125
voters	27.09	28.531	0.828
voters	32.874	18.291	0.176
voters	5.452	17.821	0.157
voters	32.039	19.406	0.016
voters	31.513	18.21	0.177
voters	79	51.466	1.125
voters	39.067	32.852	1.032
voters	18.519	17.136	0.024
voters	46.921	17.569	0.02
voters	38.627	17.979	0.172
voters	38.082	20.776	0.704
voters	45.53	20.11	0.268
voters	29.542	21.426	0.474
voters	10.033	34.443	0.906
voters	36.953	22.884	0.072
voters	38.5	23.584	0.066
voters	35.531	23.282	0.025

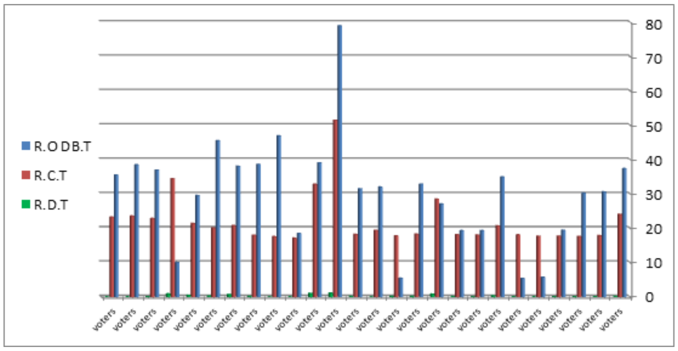
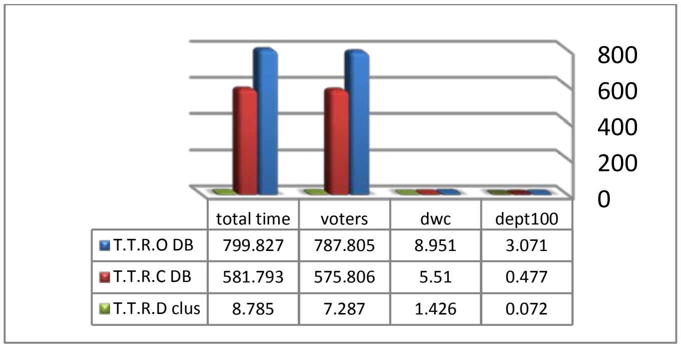


Fig. 6. Retrieval time for database voters of size 288,192 KB

The previous figures showing a comparison in the required retrieval time to retrieving from the original database, compressed database and dynamic clustering files. The following figure explains a total time for retrieving from the original database, compressed database and the file that was resulted from dynamic clustering method.



	total time	voters	dwc	dept100
T.T.R.O DB	799.827	787.805	8.951	3.071
T.T.R.C DB	581.793	575.806	5.51	0.477
T.T.R.D clus	8.785	7.287	1.426	0.072

Fig. 7. Total retrieval time

The results showed that: 1) the larger retrieving time from the original database file is 56.763 seconds and the smaller retrieving time is 5 milliseconds, 2) the larger retrieving time from the compressed database file is 34.445 seconds and the smaller retrieving time is 1milliseconds and 3) the larger retrieving time from the dynamic clustering file is 1.032 seconds and the smaller retrieving time is 1 milliseconds.

VII. Conclusion

1) In this research the dynamic clustering method was used to solve the problem of searching. Instead of searching in whole database data, the dynamic clustering enabling the user to search in data that it was relevant to the user query only. The dynamic clustering method built the clusters contained the data about the user query only. So when the user searching in this data, the results (query answers) can be received quickly much faster than searching in the whole database data or compressed database data. By this the dynamic clustering method solved the problem of the slowing in retrieving data.

2) The dynamic clustering algorithm built the clusters according to the user entries. In addition, the retrieval by using the file that is the output of a dynamic clustering process may be needed lesser inputs by the user compared with retrieval from the original database or compressed database which was needed more inputs.

3) The historical file solved the problem of building many similar copies of the same clusters. Instead of building new clusters similar to one existed, it was building the clusters only once. The user enters the query to the analyzer then the analyzer begins with searching the query in the historical file to specify if the query existed or not. This mechanism provides a saving time for retrieving answers directly if the query was existed in the historical file.

VIII. Future Works

Future work includes the proposal of new methods in order to encrypt the data and then retrieving the data from this encrypted data. The encryption of this large data in order to protect them from hackers and preservation from being stolen. As well as new methods based on artificial intelligence will be used for the retrieving purpose from large data.

References

[1] S. Parthasarathy, V. Shakila, "KNOWLEDGE CLUSTERING ON BIG DATA WITH K_MEANS ALGORITHM", International Research Journal of Engineering and Technology (IRJET), Volume: 02 Issue: 02, May-2015, e-ISSN: 2395 -0056, p-ISSN: 2395-0072.

[2] Assist.Prof.Dr.AlaaKadhim F, Prof. Dr. Ghassan H. AbdulMajeed, RashaSubhi Ali, "ICM Compression System Depending On Feature Extraction", International Journal of Emerging Trends & Technology in Computer Science (IJETTCS), Volume 4, Issue 3, May-June 2015, ISSN 2278-6856.

[3] Simranjit Kaur, RuhiBagga, "A SURVEY ON DATA MINING AND ITS TECHNIQUES", InternationalJournalInAppliedStudiesAndProduction Management, Volume1,Issue 3, 15 May- 15 August2015, ISSN2394-840X.

[4] C. J. van RIJSBERGEN, "INFORMATION RETRIEVAL", Department of Computing Science University of Glasgow,1979.

[5] E. Petraki, C. Kapetis, E. J. Yannakoudakis, "Conceptual Database Retrieval through Multilingual Thesauri", Computer Science and Information Technology 1(1): 19-32, 2013, DOI: 10.13189/csit.2013.010103.

[6] Hazem M. El-Bakry, Nikos E. Mastorakis, Michael E. Fafalios, " Fast Information Retrieval from Big Data by using Cross Correlation in the Frequency Domain", Advances in Information Science and Applications - Volume II, ISBN: 978-1-61804-237-8, 2015

[7] QIAN WeiNing, GONG XueQing and ZHOU AoYing, " Clustering in Very Large Databases Based on Distance and Density", J. Comput. Sci. & Technol., Jan. 2003, Vol.18, No.1, pp.67{76.

[8] Himali Patel, UnnatiItwala, Roshni Rana, "Survey of Lossless Data Compression Algorithms", International Journal of Engineering Research & Technology (IJERT), ISSN: 2278-0181, Vol. 4 Issue 04, April-2015.

[9] M. Premalatha, G. Baskaran, " Bootstrap Based Large Scale Data Processing Using Cluster", International Journal of Advance Research and Innovation, Volume 3, Issue 2 (2015) 359-361, ISSN 2347 – 3258.

[10] Frakes William B. and Yates Ricardo Baeza," *Information Retrieval: Data Structures &Algorithms",1991*.

[11] Jared Dean, “Big Data, Data Mining, and MachineLearning”, Value Creation for Business Leaders andPractitioners, Wiley & SAS Business Series, 2014.

[12] MansafAlam, KishwarSadaf, "Web Search Result Clustering based on CuckooSearch and Consensus Clustering", 2014.

[13] Hector Garcia-Molina, Jeffrey D. Ullman, Jennifer Widom, " DATABASE SYSTEMS The Complete Book", Second Edition, Department of Computer Science Stanford University, 2009

[14] Abraham Silberschatz, Henry F. Korth, S. Sudarshan, "DATABASESYSTEM CONCEPTS", S I XTH E D I T I ON, Published by McGraw-Hill, Copyright © 2011 by The McGraw-Hill Companies,ISBN 978-0-07-352332-3.

[15] Dr. Pushpak Bhattacharyya, JoydipDatta, "Ranking in Information Retrieval", April 16, 2010.

[16] NamrataGadkari, Sylvester Savio Raj, HarshadRaka, "Query Subtopic Mining from Search Log Data", International Journal of Current Engineering and Technology, Vol.5, No.3 (June 2015), E-ISSN 2277 – 4106, P-ISSN 2347 – 5161.

[17] Tara Guthrie, "BOOLEAN SEARCHING", 2010.

[18] Ceri, S., Bozzon, A., Brambilla, M., Della Valle, E.,Fraternali, P., Quarteroni, S., " Web Information Retrieval", 2013, ISBN 978-3-642-39314-3