Communications on Applied Electronics (CAE) – ISSN : 2394-4714
Foundation of Computer Science FCS, New York, USA
Volume 4– No.8, April 2016 – www.caeaccess.org

Applying Local Optimization Algorithms in Clustering Combination with Diversity Maximization

Z. Faizal Khan, PhD
Department of Computer and Network Engineering,
College of Engineering, Al Dawadmi,
Shaqra University, Kingdom of Saudi Arabia

Mohammed Khaleel Anwar
Department of Electrical Engineering,
College of Engineering, Al Dawadmi,
Shaqra University, Kingdom of Saudi Arabia

ABSTRACT
Automated searching needs is an effective key for the development of cryptography algorithms. The retrieval of encrypted contents from a huge set of distributed databases leads to complexity in the grid computing environment. This paper is based on a search of some set of keywords in grid computing environment. Most of the key based cryptography algorithms developed by earlier researchers are more effective but each and every algorithm has its own disadvantages. Majorly, there arises a huge need to direct the data contents to Grid computing system in the form of cryptic. No algorithm in an intelligible advanced form has been available to the researchers. Complexity and computational complexity present in these algorithms make the retrieval task more inefficient. In order to overcome these drawbacks, we proposed a novel fuzzy based optimization technique in the Grid computing environment based on an encrypted text along with Fuzzy rules. We obtained a satisfactory result in keyword search using this novel method. The result obtained was most effective and the retrieval time is very much reduced in the Grid computing environment.

Keywords
Cryptography algorithms, computational complexity, scheduling, Grid Computing Environment, Fuzzy logic, peer to peer search system.

1. INTRODUCTION
With the rapid growth of data and Computational needs, distributed Systems and Computational Grids are becoming recent trends in research nowadays. The huge amount of computations in a Grid [16] environment cannot be performed even by the best algorithms. However, its performance can still be improved by making sure all the resources available in it by proposing advanced algorithms. Computational Grid [12] is a well-established platform that gives assurance to provide a vast range of heterogeneous resources for high performance computing. Efficient resource management based systems and Grid job scheduling [14] are the key factors that are present in order to optimize the use of these resources

Data encryption scheme makes effective utilization of data present in it. The data encryption is a challenging task because there might be a large amount of outsourced data files present in it [2]. In Grid computing environment, data owners may share their important data with a large number of users can access it. And they want to retrieve a certain specific and important data files during a particular time. One of the most popular and easiest way to do retrieve the data is through keyword-based search mechanisms. Such advanced keyword search algorithms paves the batter way for users to retrieve files selectively present in it and this methodology has been widely applied in plaintext searching scenarios. Unfortunately, data encryption restricts the user's ability to perform any keyword based search and further demands the protection of keyword privacy. It also makes the traditional plain text search methods unsuccessful for encrypted data present in the Grid environment.

Traditional searchable encryption schemes given in [5], [7] [11], [9], [18] allows an individual user to search over the encrypted data throughout the keywords. This could be done without decrypting it. These various techniques support only the conventional Boolean keyword search without accounting the any relevance of the files present in their search result. In this paper, we propose a novel fuzzy logic keyword search technique in the Grid computing environment in order to overcome the drawbacks present in the existing works.

2. LITERATURE SURVEY
The classical work proposed by Goldreich et al on oblivious RAMs can resolve the problem of doing private searches on remotely encrypted data's [10]. Although their scheme is more efficient in nature and nearly optimal, it does not appear to be efficient in practice as large constants are hidden in it. They proposed a scheme which encrypts each word or each pattern of a document separately. Their approach has the following disadvantages such as it is not compatible with existing encryption mechanisms. For that, a specific encryption scheme must be used. It cannot process the compressed data effectively.

Kiyohide Nakauchi developed a peer to peer search system in order to increase the possibility of discovering the desired data items and key mechanism during the process of query expansion. The query is expanded based on the relationships between the keyword present in it [13]. These keyword relationships are improved through various search and retrieval process and each of them is shared among nodes holding similar data sets present in it. Jayalatchumy.D et al [11] developed a novel search engine which is built on top of the Grid environment with the help of various software agents. Their proposed search engine allows the user to search through various resources which are stored in geographically distributed digital collections.

Changjiang Hou et al [6] presented an efficient public-key encryption with keyword search scheme from lattices present in a set of keywords. Pallis.G et al [15] addressed the unavailability of software-related metadata on Grid sites. The authors developed a method for the already available Minersoft, a Grid harvester that visits Grid sites, receives their file-systems, identifies and classifies software resources, and

discovers implicit associations present in between them. In their work, the authors presented the design and implementation of the core information retrieval [17] component as a component of Minersoft - the Software Graph in order to achieve a high search efficiency.

Ayad Ibrahim et al [1] focused on constructing a flexible secure index that allows the cloud server to perform the approximate search operations. The authors employed Order Preserving Symmetric Encryption (OPSE) to protect their keywords. First their scheme divides the candidate list in terms of secure pruning codes. The next step uses a semi honest third party to determine the best matching keyword depending on secure similarity function. The authors tested their framework by real dataset to check its correctness and security. [1]

3. FUZZY LOGIC BASED PARSER

A Fuzzy logic based parser is basically a filter connected to an inference table in a database. The inference table collects all words which are not relevant to a particular set of words. Basically they are text based words. A table is already present in the database which has records of unwanted keywords which are mostly verb, articles, prepositions, conjunctions etc. Each and every word from the given text file is allowed to map the unwanted keywords present in the table through the key word search algorithm. All those mapping words are eliminated from the file and the left over words are basically the keywords which were um matched with the words present in the database.

When the fuzzy based parser finds an input document for the process of extraction of keywords first it reads the complete document word by word. Every word of the document is being compared with the table containing unwanted less sensitive words. If a matching word is found during the process of comparison, the word gets eliminated automatically. If not, it is accounted as the presence of a keyword for further processing. Each keyword is further indexed with a uniform identity number which in turn associates the original file during the process of indexing. Repeat of key words removed in this process. The output will be many to one mapping such as the document is associated with few keywords. Another option allows a direct mapping to the file, without indexing process. The fuzzy based inference table will have an entry of file id, followed by the extracted keyword. This process will continue for a number of keywords present in the total file.

3.1 Keyword Search process

In Grid computing environment, there is a need to divide and distribute the larger tasks in to various grid instances so as to enable to work concurrently in order to solve a problem. These concurrent processes save a lot of time and enhance quality in job execution. Whenever a job is submitted to a grid controller, it then takes job and does the parsing process. It is a kind of forward processing where in the output is a parser table consisting of keywords with their corresponding file name. As grid instances do not engage much in applying independent algorithms to modulate the data into an intelligent form, there is a certain percentage of requirements attached with the grid controller to take care of this specific task. The grid controller modulates the text with a certain specific public keys [3]. And the modulated unintelligible form of text is being splitted into n number of components. And these components are assigned to grid instances for further processing.

3.2 Inference

The inference is a process of forming a new decision based on the existing information [4]. In this work, inference is carried out in order to form a membership table shown in table 2 by assigning output values based on feature values present in Table 1. At the end of inference, the outputs as keywords are obtained. The numbers in parenthesis in Table 1 are the values assigned based on the fuzzy rules shown below:

For each input keyword

If Chr(a) = 5 & chr (b)==7

then

Chr(c) = Chr(a)+5

ENDIF

else

Chr(c) = Null.

C Key word search algorithm

The grid controller has a fuzzy inference table which is being matched with the keyword text supplied by the user. If matching doesn't occur by the fuzzy rules, then the grid controller without contacting the grid instances directs output messages to the user stating non availability of text.

Based on the fuzzy rules, if it is found then the grid controller knows the mapped file from the parser table, then it generates a request to all the grid instances to supply the files. Every grid instance now looks for the file being demanded by the grid controller. If grid instance has a file, it passes the part of file along with a segment number. Every grid does the same process throughout the entire process. The fuzzy controller receives all the total text along with their consequent numbers which are intelligible form of the complete document. The fuzzy based controller also does the reverse process of decryption with the key which is already provided to it .A message digest can also be used to ensure user's authenticity between the fuzzy based grid controller and the user. Thereby the security between the user and the fuzzy based grid controller is incorporated. The entire scenario is given in the below algorithm

Algorithm

For each input keyword

If find (Text in table) is false

Display message "THE TEXT NOT FOUND"

Return

Else

Display message "THE TEXT FOUND"

End for

For each file found

Map inference table to get character string of sequence S

1) Map the sequence S into a block having four columns and N/4 rows in the fuzzy inference table.

2) Perform Column shift and Row shift in a certain specified order to the resulting symbol block.

3) Perform Prime Diagonal shift and Secondary Diagonal shift in a certain specified order to the block.

4) Represent the outcome in a linear order to get the encrypted text.

5) Perform Secondary Diagonal shift in an order carried out in step4 to the Block obtained in step5.

6) Perform Primary Diagonal shift in an order carried out in step3 to the Block obtained in step7.

7) Perform Row shift in an order carried out in step 3 and Column shift in an order carried out in step 2 to the Block obtained in step 8.

8) Formulate the outcome in linear array.

9) Apply substitution scheme to the linear array to get the decrypted text.

End for

Following fuzzy feature table has been used for effective encryption scheme as a substitution where an intruder cannot be able to make any kind of guess.

Table 1: Fuzzy Feature Variables Table

1	A	Chr(175)	16	P	Chr(225)	31	4	Chr(165)	46	j	Chr(15)
2	B	Chr(177)	17	Q	Chr(226)	32	5	Chr(164)	47	k	Chr(16)
3	C	Chr(182)	18	R	Chr(227)	33	6	Chr(162)	48	l	Chr(17)
4	D	Chr(184)	19	S	Chr(228)	34	7	Chr(155)	49	m	Chr(18)
5	E	Chr(187)	20	T	Chr(229)	35	8	Chr(149)	50	n	Chr(19)
6	F	Chr(191)	21	U	Chr(230)	36	9	Chr(139)	51	o	Chr(20)
7	G	Chr(216)	22	V	Chr(231)	37	a	Chr(2)	52	p	Chr(21)
8	H	Chr(217)	23	W	Chr(232)	38	b	Chr(3)	53	q	Chr(22)
9	I	Chr(218)	24	X	Chr(233)	39	c	Chr(4)	54	r	Chr(23)
10	J	Chr(219)	25	Y	Chr(234)	40	d	Chr(5)	55	s	Chr(24)
11	K	Chr(220)	26	Z	Chr(235)	41	e	Chr(7)	56	t	Chr(25)
12	L	Chr(221)	27	0	Chr(172)	42	f	Chr(8)	57	u	Chr(26)
13	M	Chr(222)	28	1	Chr(171)	43	g	Chr(236)	58	v	Chr(27)
14	N	Chr(223)	29	2	Chr(167)	44	h	Chr(12)	59	w	Chr(127)
15	O	Chr(224)	30	3	Chr(166)	45	i	Chr(14)	60	x	Chr(128)
61	y	Chr(134)	62	z	Chr(135)	CODE SHEET FOR SUBSTITUTION					

The Fuzzy Feature variables table shown in Table 1 is used to get back the text present in the given input. The keywords are extracted with the help of fuzzy logic and one to one mapping. The Fuzzy rules are applies with the mapping and it is generated between the keywords extracted and the names of files as an ordered pair as (XXX, YYY) where XXX stands for the keyword and YYY stands for the name of files. These variables are combined at the time of file indexing but before the encryption of texts.

4. RESULTS & DISCUSSION

In this paper, Inference is used to generate the fuzzy rules for searching the keywords in the grid computing environment It has been experimentally verified that an average of 25 percent keywords are extracted from each document. Thereby 25% of words alone are to be matched for retrieving files containing the keywords using this fuzzy based word search. Proposed algorithm covers more number of files and searches the keywords in lesser time since the fuzzy logic is incorporated with it. But the Ranked Keyword Search algorithm [10] and earlier algorithm [8] takes more time to search and the number of files searched is lesser in comparison with the proposed fuzzy logic based searching algorithm. The following membership table shows the no of keywords extracted from the encrypted text file.

Table: 2 Membership Table From The Encrypted Text File

File No	Encrypted File Size	Groups	Keywords in Each Group
1	34 KB	{1,2,3}	130
2	33 KB		
3	32 KB		
4	33 KB		

5	32 KB	{ 4,5,6}	140
6	33 KB		
7	33 KB	{7,8,9}	127
8	33 KB		
9	56 KB	{ 10,11}	183
10	55 KB		
11	55 KB	{12,13}	161
12	55 KB		
13	56 KB	{14}	145

5. ACCURACY ANALYSIS

Following are the measuring factors that are taken up for considering the performance of this proposed work. The factors are Accuracy and Search time consumption.

Accuracy: The fuzzy logic used in this paper had given better accuracy. The performance of our experiment using software fuzzy logic yields better frequency than existing method [8]. Average accuracy achieved in this proposed system is 75.32 % whereas frequency vector coding's average accuracy is 68.7% and the existing methods accuracy is 70.57 % respectively.

Searching Time Consumption: The average time of search for the proposed work is 1.0132 ms. Performance has been depicted in the following graph. The proposed algorithm is compared with the existing algorithm like Ranked Keyword Search algorithm and existing method.

The keyword search and its performance are directly proportion to the sensitivity of it. That is, when there are more number of keywords we mean the document is more sensitive in nature. This will cause extraction of more keywords by fuzzy logic and thereby eliminating fewer number of unwanted words vice versa. By changing the sensitivity of the input document, the performance of the proposed methodology can be improved.

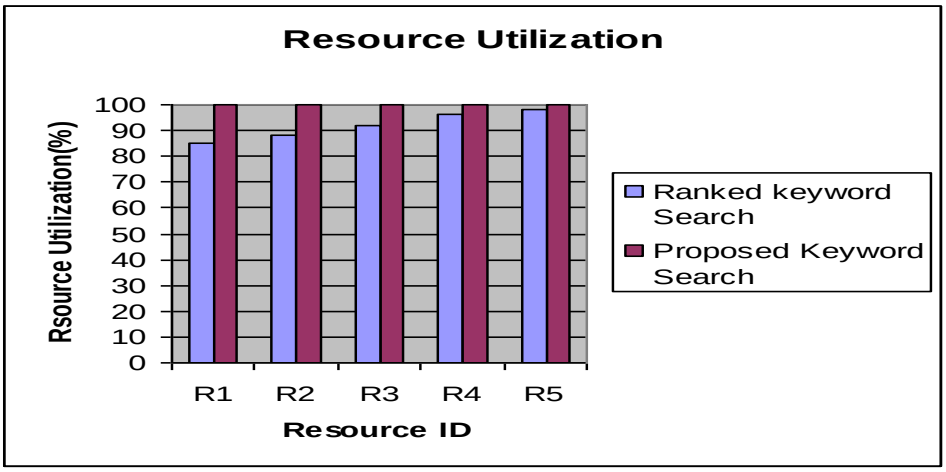

Figure 1: Resource utilization graph

6. CONCLUSION

Keyword search is a common phenomenon in Grid computing, such searching mechanism may not be possible in encrypted documents which are in intelligible form. There is a need for a novel mechanism to map words and extract the relevant documents. It is a challenging task in grid computing. Obtaining a better accuracy and reduction of time consumption is another prime objective in key work search. The proposed methodology improves the performance of keyword searching by introducing the fuzzy logic based keyword search and reducing the keyword search time.

7. REFERENCES

[1] Ayad Ibrahim, Hai Jin, Ali A. Yassin, Deqing Zou , "Approximate Keyword-based Search over Encrypted Cloud Data", IEEE International Conference on e-Business Engineering, pp 238 -245,2012.

[2] Boldyreva.A, Chenette.N, Lee.Y, and O'Neill.A.,"Order-Preserving Symmetric Encryption ", Proc. International Conf. Advances in Cryptology (Eurocrypt '09), 2009.

[3] Boneh.D, Crescenzo.G.d, Ostrovsky.R, and Persiano.G.,"Public Key Encryption with Keyword Search," Proc. International Conf. Advances in Cryptology, 2004.

[4] Faizal Khan, Z & Kannan, "Intelligent Approach for Segmenting CT Lung Images Using Fuzzy Logic with Bitplane", Journal of Electrical Engineering and Technology, Vol. 9, No. 4, pp-742- 752, 2014

[5] Chang.Y.C and Mitzenmacher.M, "Privacy Preserving Keyword Searches on Remote Encrypted Data ", Proc. International Conf. Applied Cryptography and Network Security, 2005.

[6] Changjiang Hou, Fei Liu , Hongtao Bai , Lanfang Ren ,"Public-Key Encryption with Keyword Search from

Lattice", Eighth International Conference on P2P, Parallel, Grid, Cloud and Internet Computing, pp 336-339,2013.

[7] Cong Wang, Kui Ren, "Enabling Secure and Efficient Ranked Keyword Search over Outsourced Cloud Data ", IEEE transactions on parallel and distributed systems, Vol. 23, No. 8, pp 1467-1479 August 2012.

[8] Dr. Z. Faizal Khan, Dr. S. Veeramalai and J. Velmurugan, "Automatic Keyword Search using Encrypted Text in Grid Computing Architecture", Journal of Applied Sciences Research, Vol 11, issue 14, pp- 190-194, September 2015.

[9] Goh.E.J , Secure Indexes," Technical Report 2003/216, Cryptology ePrint Archive, http://eprint.iacr.org/, 2003.

[10] Goldreich.O and Ostrovsky.R, "Software Protection and Simulation on Oblivious Rams," Journal ACM, Volume 43, No. 3, pp. 431-473, 1996.

[11] Jayalatchumy.D , Kadhirvelu.D, Ramkumar.P , "Design of an Intelligent Answering System Through Agent Based Search Engine Using Grid Technology" First International Conference on Emerging Trends in Engineering and Technology, pp 519-524,2008.

[12] Kyriaki Z.Gkoutioudi, Helen D.Karatza, "Multi-Criteria Job Scheduling in Grid Using an Accelerated Genetic Algorithm", Journal of Grid Computing, Volume 10, No 2, pp 311-323, June 2012 .

[13] Kiyohide Nakauchi, Yuichi Ishikawa, Hiroyuki Morikawa, Tomonori Aoyama,: Peer-to-Peer Keyword Search Using Keyword Relationship, 3rd IEEE International Symposium on Cluster Computing and the Grid, pp 359-366, May 2003.

[14] Lee.Y.H, Leu.S, and R.-S. Chang, "Improving Job Scheduling Algorithms in a Grid Environment," Future Generation Computer Systems, volume 27, pp. 991-998, 2011.

[15] Marios D. Dikaiakos, Asterios Katsifodimos, George Pallis. Miner soft: Software Retrieval in Grid and Cloud Computing Infrastructures." ACM Transactions on Internet Technologies. Volume 12, No 1, June 2012.

[16] Simone A.Ludwig, Azin Moallem,"Swarm Intelligence Approaches for Grid Load Balancing", Journal of Grid Computing, Volume 9, No3, pp 279-301, September 2011.

[17] Singhal.A, Modern Information Retrieval: A Brief Overview, IEEE Data Engineering, Volume 24, No 4, pp 35-43, 2011.

[18] Zerr.S, Olmedilla.D, Nejdl.W, and Siberski.W, Zerberr, "Top-k Retrieval from a Confidential Index", Proceedings International Conference Extending Database Technology: Advances in Database Technology, 2009.