An Arabic Web search engine using grid computing and Artificial Intelligence techniques

Mohammed Mahmoud Ibrahim Sakre

Abstract—This research is considered as a result of an accumulative work for many years to demonstrate a model for the heavy computational components of any World Wide Web (WWW) search engine. This architecture is based on the grid computing. The crawling load is distributed over a set of computers to retrieve more crawled pages in less time. The proposed architecture of the indexer distributes the indexing load over a set of computers and supports the dynamic indexing to deal with the frequent changes in the web contents. So, the proposed architectures of the crawler and the indexer support the freshness of the web pages. The used freshness technique is considered in the crawler and the indexer where the dynamic indexer is responsible of determining the old pages and sending them to the crawler to revisit them for updating. The Search module is implemented including Arabic morphological analysis/generation and synonym dictionary which are combined to produce an Intelligent Arabic Internet Search module. The use of these linguistic tools is proved, experimentally, to have positive effects on both Precision and Recall measures where the average precision exceeds the value of 0.92. This design is implemented for Arabic language but it suits any other language with language-related modifications.

Index Terms— Grid Computing, Internet Search Engine, Crawling, Indexing, Artificial Intelligence, Natural Language processing.

———————————— ◆ ————————————

1 INTRODUCTION

Rcently, the World Wide Web (WWW) is one of the main sources of information for a large number of people. WWW search engines are considered as the mediators between online information and people. WWW search engines require computers with high computation resources for processing to crawl web pages and require huge data storage to store billions of pages collected from the WWW after parsing and indexing these pages. The proposed key for this problem is offered by the use of the Grid Computing. The Grid computing term recognized in the mid of 1990s and it refers to a proposed distributed computing infrastructure [1,2].

The typical design of any search engine consists of three stages in which a Web crawler creates a collection of pages which is indexed and searched. This model, in which operations are executed in strict order: first (Crawling, then indexing as pre-processing phases), and then (searching as a run-time phase) is explained in figure (1). [3,4].

The crawling starts with a set of URLs to fetch their pages and parses them to extract the new URLs exist in these pages. Each extracted URL is either a new discovered URL which should be visited next [4], or an old URL for which the weight of its page should be increased. This will affect the page rank during the searching stage.

The indexing stage operates on the pages collected during the crawling stage. It parses the pages and generates the inverted index as has been described in a previous research of the author and others [5].

The searching stage gets answers to the users' queries based on the non-stop words of the query terms. Freshness of the web pages is an important factor that affects the efficiency of the search engine. There are different techniques to keep the web pages up-to-date [6].Page Ranking is the process which estimates the quality of a set of results retrieved by a search engine and presented to the user. Search engines have taken a lot of effort to rank Web objects and to retrieve the correct and desired information contained on the data bases of the WWW. Freshness and Page Ranking topics are considered in the proposed model of this research.

The searcher uses the indexed database to find the proper web pages which contains an answer to the user query. The search results are ordered according to their relativity to the query using the page ranking parameters, calculated during the execution of the crawler and the indexer, and presented to the user. The search engine modules of different search engines differ from each other by the way of working. Some search modules use the query words as they keyed in by the users. Another search engine give the user the ability to use Boolean functions. However, more advanced search engines perform some lexical and/or morphological analysis on the keywords like the one presented in this research for Arabic language.

There are a number of research groups that have been working in the field of distributed computing. These groups have created middleware, libraries and tools that allow the cooperative use of geographically distributed resources unified to act as a single powerful platform for the execution of parallel and distributed applications. This approach of computing has been known by several names, such as metacomputing, scalable computing, global computing, Internet computing and lately as grid computing [1], [2], [7].

Alchemi system is an open source software toolkits developed at the University of Melbourne, which provides middleware for creating an enterprise grid computing environment. Alchemi consists of two main components are manager and executer. More than one computer runs the executor program and only one computer run a manager which stores the address of the executors which are connected to the manager. The manager receives the threads from the client application and distributes these threads over the executors connected. The manager stores the execution location and time of execution of each thread. [22] In this research Alchemi tool is used to implement the proposed model.

2 RELATED WORK

- M. Sakre and others [14] presented powerful algorithms, like the basic modules of internet search engine and Arabic morphological analyzer/generator which are combined to produce an experimental prototype Intelligent Arabic Internet Search Engine. Proposal does not depend on distributed computing. They declared the need for many Arabic linguistic tools to be developed.
- M. Sakre and others [4] presented a search engine crawler using grid computing. They used Alchemi as a grid computing tool in their implementation.
- M. Sakre and others [5] presented the distribution of the indexing process over a cluster of computers in grid computing which improves the performance. They used Alchemi as a grid computing tool in the model implementation. They considered the dynamic indexing in their work.
- Cambazoglu et. al [8] [9] presented the architectural design issues and implementation details of a Search Engine for South-East Europe (SE4SEE) which is a socio-cultural search engine running on the grid infrastructure. The main goal of SE4SEE is to resolve the page freshness problem by performing the search on the original pages residing on the Web, rather than on the previously fetched copies as done in the traditional search engines. SE4SEE also aims to obtain high download rates in Web crawling by making use of the geographically distributed nature of the grid.
- Kaur et al [10] gave a review on evolution of cloud computing, its comparison with grid computing and various approaches to cloud computing.
- Rajshri S. Patill et al. [11] explained how data replication is applied to reduce data access time and explained that the file replication has an effective functionality in Data Grid that not only minimizes total access time by replicating most accessed data file at appropriate location but also improve data files availability in a grid environments.
- Fareed, N.S. et al. [12] introduced a proposed design for an Arabic Question Answering system based on Query Expansion ontology and an Arabic Stemmer. Improved results obtained using AWN as a semantic Query Expansion and Khoja stemmer as a stemming system.
- Haya et. al [13] proposed a tool named Grid Search And Categorization Engine (GRACE) that allows users to search through heterogeneous resources stored in geographically distributed digital collections. The GRACE toolkit will also provide a categorization engine which will dynamically integrate and categorize results from the various data sources. The categorization engine will be based on Automatic Idiomatic Representation (AIR) technology which is designed by Virtual Self, one of the GRACE partners. Results can be automatically categorized regardless of how they are formatted or whether they contain metadata. Moreover, the AIR technology is language independent, so with the aid of language lexicons it will work on sources in various languages. To begin with, GRACE will be capable of automatically identifying and then categorizing results in the following languages: English, German, Swedish and Italian. Additional languages may be added at a later stage.

3 SEARCH ENGIN MODULES

In this section descriptions of the proposed search engine modules, namely the Crawler; the Indexer and the Searcher, are presented.

3-1 The Crawler

The aim of the Crawler is to collect Web pages and extract the contained URLs in these pages and visit the new pages to create a large repository of Web pages. The crawling process consists of multiple processes in which requests for pages are sent and pages are received as responses of these requests. The received pages are parsed to extract any URL contained in them. The new URLs are added to the URLs Queue, and so on.

The Crawler architecture is based on the concept of parallel / distributed processing where the goal is to reduce the overall processing time of collecting large amount of pages and extracting the embedded URLs. The used software is classified into the utility software of the grid computing and the application programs under discussion like the Crawler. The application programs require intense computing actions which could be divided into subtasks that could run in parallel and combined later to yield the desired result. These sub-tasks are executed in different machines in order to ensure time effectiveness and resource utilization [14]. So, the crawling processes are distributed into threads and multiple computers are used to execute the threads with balancing to distribute the efforts and minimize the time of the crawling execution. Figure(2) presnt the proposed architecture of the Crawler [4]. This module is Implemented using Alchemi Software Development Kit (SDK) that includes a Dynamic Link Library (DLL) which supports multithreaded applications.

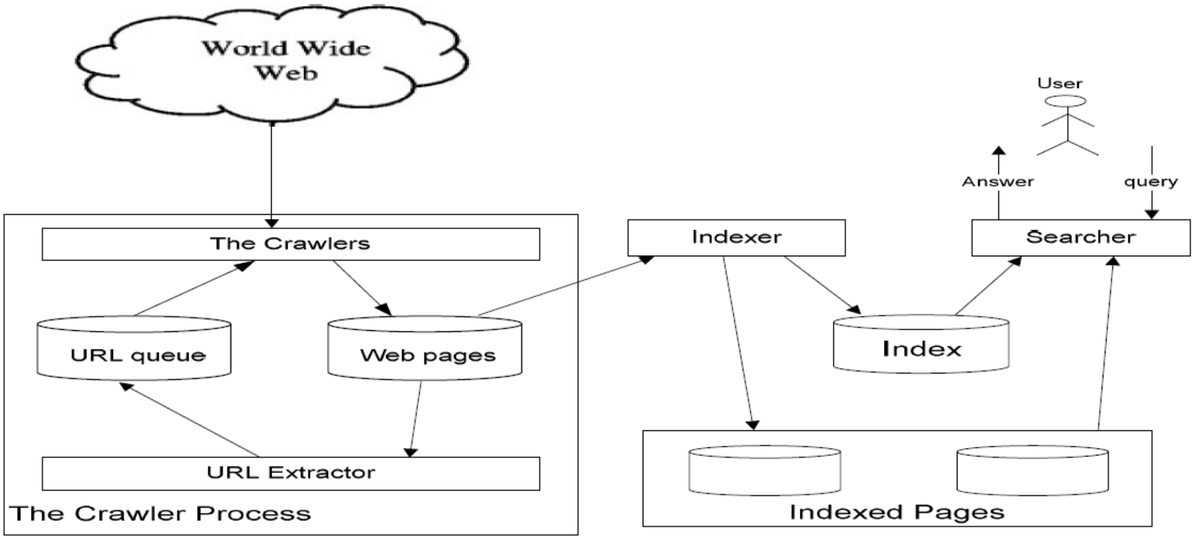

Fig 1 Search Engine Generic Architecture

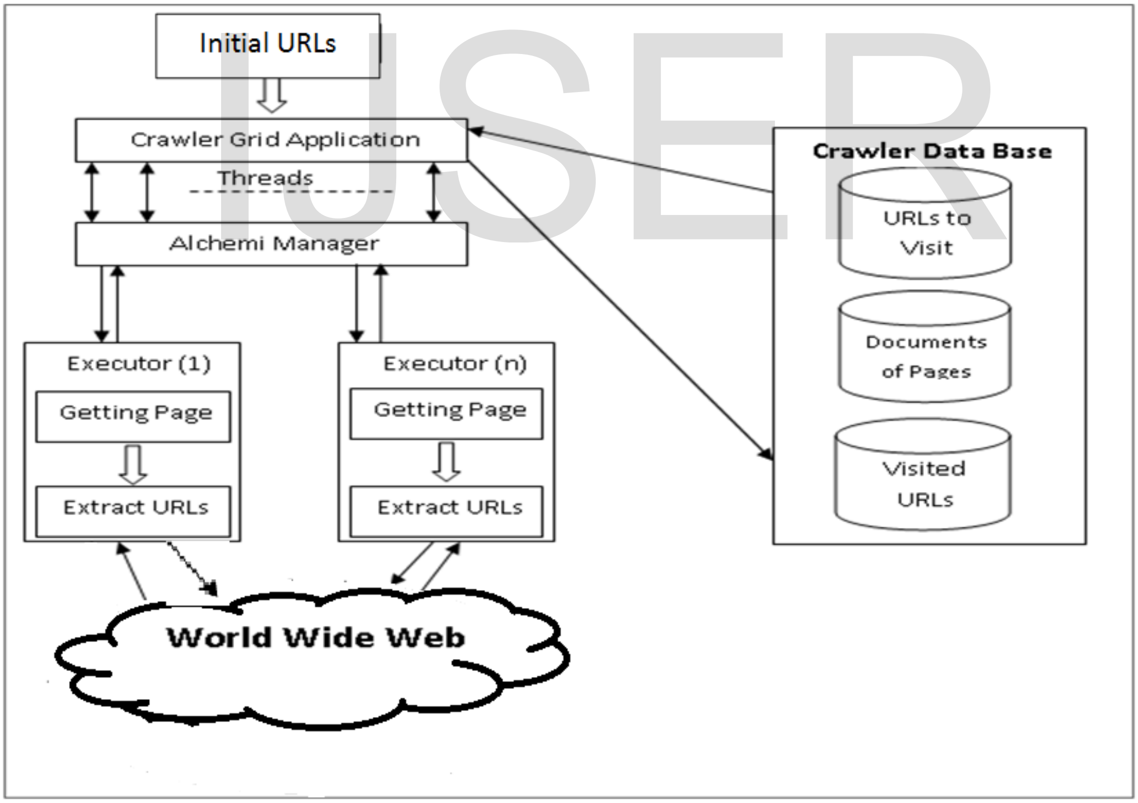

Fig 2 The Crawler architecture.

3-2 The Indexer

The indexer operates on the data base of pages of the WWW resulted as the output of the crawler. The efficiency of the index affects the time required to retrieve the required pages for the user queries. This mean that the used index technique affects the searcher performance. The proposed indexer structure focuses on three main issues:

- The first issue is the proper index structure which will have a better effect on the searcher performance.
- The second issue is the index distribution, since the WWW contains billions of pages [15] which mean a very huge index for the WWW to be created and stored.
- The third issue is the dynamic indexing that trace changes in WWW content.

For The first issue, a single- level ordered secondary index based on key words is constructed to speed up the search [16]. Figure (3) shows the main tables used in the indexed Database, It shows also how the index is constructed. The indexer sub-program in this system has many functions:

- It extracts the Title from each page and adds it in the proper row of the page-table.

- It extracts the key words from each page and adds them to the Index-table.

- It extracts the description for each page and adds it in the corresponding row of the page-table.

For the second issue, the index creation and distribution is achieved by distributing computing and storage loads over a set of computers. This is designed to be implemented under the control of one master computer which manages a set of computers and distribute the workload of extracting indices over them. The index lists are returned to the master computer who distributes them according to alphabetic order of the key terms over another set of computers. Each of these computers has a large storage to contain a part of the index repository.

Regarding the third issue, the dynamic indexing can be achieved by using a secondary (temporary) index beside the main index. The main index contains the index data that have been collected from the beginning and before merging the secondary index. The secondary index is responsible of indexing the pages that has been crawled while running the searches. The crawled pages may be new pages or may be old but have been updated. The secondary index generates indices for these pages and stores them temporarily. When the secondary index reaches a predefined limit, it is merged with the main index [5]. Figure (4) explains the operations and the structure of the indexer. This module is implemented using Alchemi Software Development Kit (SDK) as well.

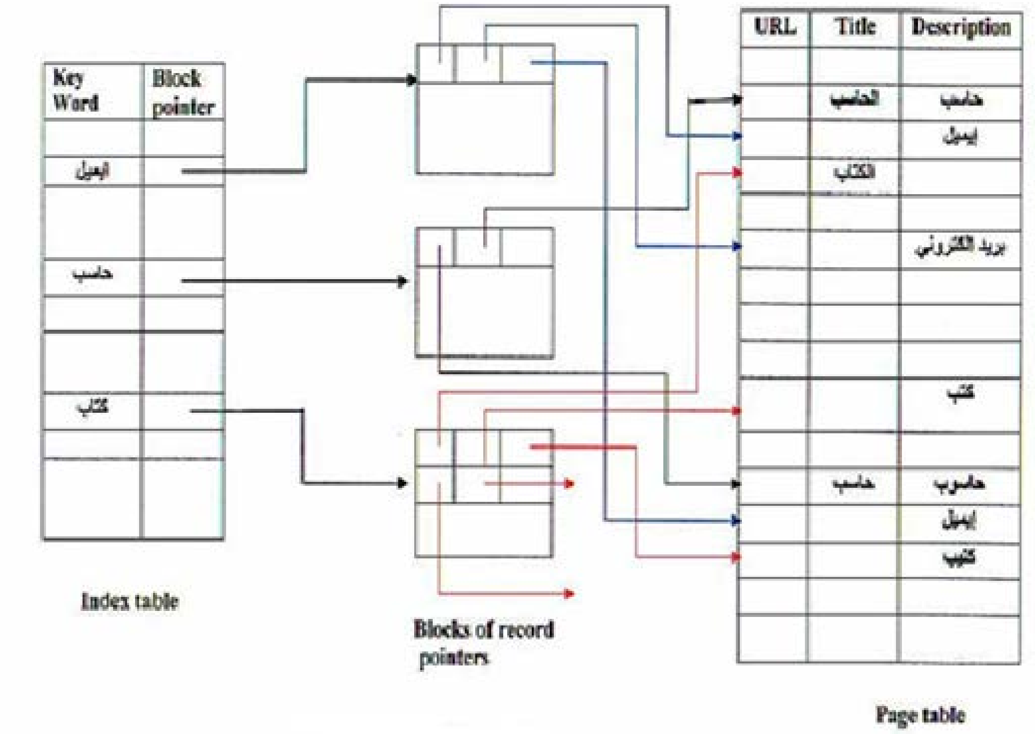

Fig 3 A single-level ordered secondary index based on keywords field.

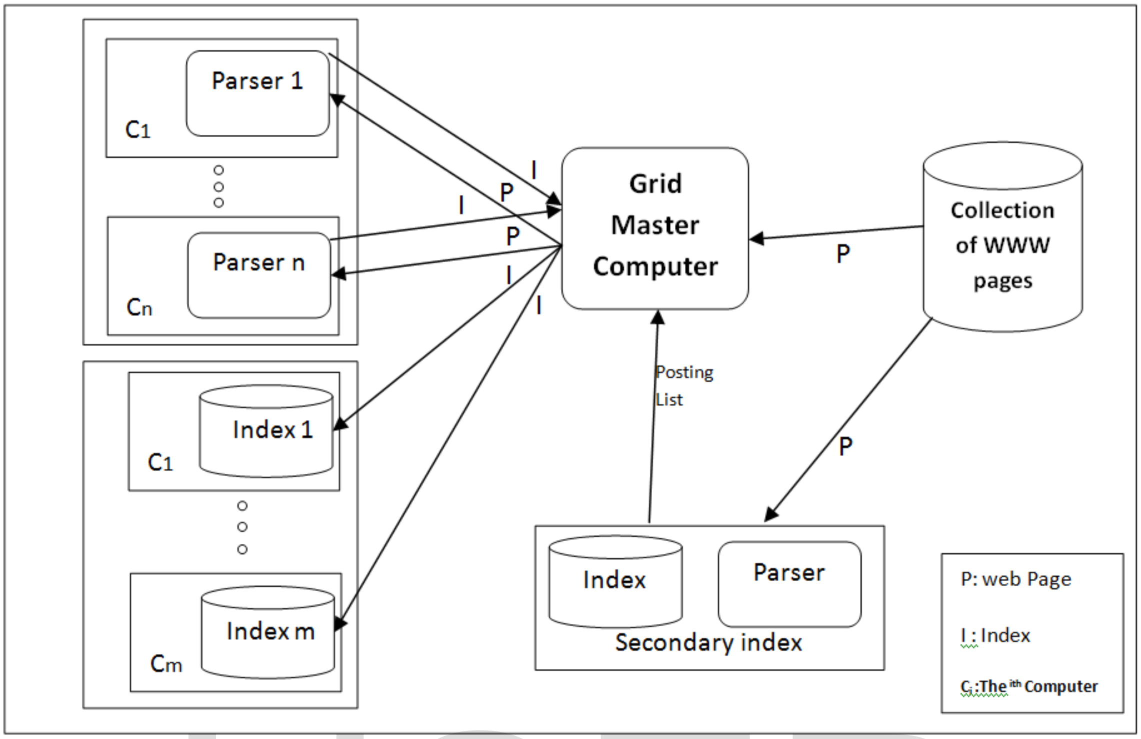

Fig 4 The structure of the dynamic distributed index.

3-3 The Searcher

The search module in any Web search engine represents the interface between the user and the search engine.

The searcher module of the proposed Arabic search engine enables the user to find the web-pages related to his query in four searching modes [17]. These are:

- **Normal search:** In this mode, the search module seeks for the web-pages which contain both the exact word(s) of the user's query.

- **Search with words and synonyms**: In this mode, The search module looks up a Synonym Dictionary for keywords' synonyms and uses both of them to find the related web-pages.
- **Morphology based search**: The system contains a morphological module which generates the related words, that have the root (or stem) characters of each keyword and select those which are related to the meaning of the original keywords. So, the number of key words will increase which yield to an increase in the number of resultant web-pages which means expecting better Recall but not necessarily better Precision.
- **Synonyms and Morphological search**: This makes use of all the advantages of the above mentioned search modes.

Figure (5) shows the main components of the search module of the proposed Arabic search engine including the use of morphological processing and synonyms of words of t Sakre M. et al, he Arabic queries. Prolog is used as a powerful tool to accomplish this linguistic task [18], [19].

During the Morphological analysis, each Arabic keyword in the query is morphologically analyzed in three steps to find its root. These steps are [18]:

1. Stripping the prefixes from the word:

This is implemented in Prolog by defining a set of predicates for the Arabic prefixes like (إست , ن , سن ....).

The general form of these predicates is:

Strip (Key_word, Rest).

Example of this predicate is:

Strip ([‘إ’,’س’,’ت‘,Rest], Rest).

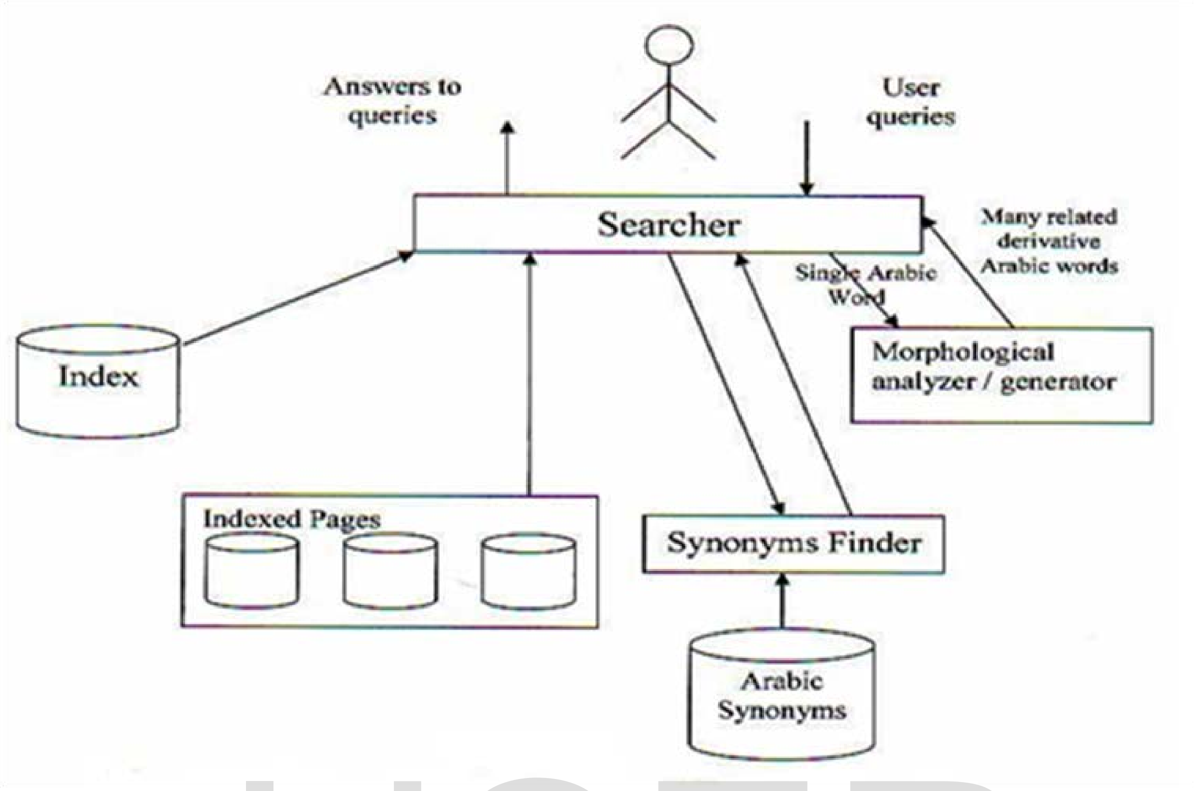

Fig 5 The main components of the search module in the proposed Arabic search engine

2. **Stripping the postfixes from the word:**

This is implemented in Prolog by defining a set of predicates for the Arabic postfixes like (ت, ك ,كم ,هما , هم, ...,).

The general form of these predicates is:
Strip_reverse (Key_word, Rest).

Example of this predicate is:
Strip_reverse ([Rest,'ه','م','ا'], Rest).

3. **Finding the root from the rest of the word:**

This is implemented in Prolog by defining a set of predicates for all the possible Arabic weights of derivatives, "المشتقات ",

and the corresponding roots. The general form of these predicates is: Weight(Rest_of_Arabic_word, Root).
Example of these predicates are:

Weight(['X','Y','Z'], ['X','Y','Z']) ….Three-letters root.

Weight(['إ','ن',X','Y','Z'],(['X','Y','Z'])….Three-letters root.

Weight(['ت','X','Y','Z','W'],(['X','Y','Z','W'])….Four-letters root.

The non-deterministic feature of prolog is used to generate all the derivatives of the resultant root. A semantic dictionary, which defines a semantic category for each Arabic word, is consulted to select only words with the same semantic category like that of the original keyword. The selected words are added to the list of keywords and a normal search is performed on all the members of the list.

Finally the three modes (Normal search , morphological search and synonyms search) are offered separately or combined to give the user freedom to select any of them or all of them in his internet search.

Ranking the resultant Web pages is out the scope of this research at the present stage. For simplicity the resultant Web pages are ordered according to number of query key words contained in each page and the number of repetition of each keyword.

When the searcher receives a user query, it analyzes it to extract the needed keywords according to the mode selected by the user as mentioned above. Then it sends the resultant keywords to the master grid computer which contains a software program responsible for publishing these keywords of the query to the main and the secondary indices and merges the received results from them together to get the result with the latest Web pages updates. Therefore any query's keywords received from the searcher of the search engine to the grid master computer; the master computer will send the query's keywords to the proper computer of indices to search in its index for the query's keyword and find the related information. This distribution of the index according to keywords, as mentioned before, has a positive impact on the performance of the searcher of the search engine. Figure (6) shows the structure of the proposed search engine with the dynamic index and the relationship between all of the modules of the search engine. The secondary index works while running the system and receives keywords of queries from the master grid computer to find their indices for the searcher.

4 RESULTS AND EVALUATION

Due to the limited capabilities of software and hardware, each module of the proposed system is evaluated separately using six personal computers. The most powerful computer acts as the Grid Master Computer where Alchemi executor program is installed. The other computers work as executers for the grid and are connected to an internet Line. Both of the Crawler and the Indexer are evaluated on this test bed.

4-1 The Crawling Evaluation:

The crawler grid application runs on the computer which contains Alchemi manager. Several experiments has been run, the first is implemented with one executer. The second is implemented with two executers and so on up to five executers. Each time 50, 100, 150, 200, 250, 300 pages are collected and the execution time for each run is observed. Figure (7) reflects the effect of using more computers, as execution nodes, on the time and the number of crawled pages [4].

4-2 Indexing Evaluation :

The mentioned above test bed was used to evaluate the indexer. The indexer grid application runs on the computer which contains Alchemi manager. Several experiments has been run, the first is implemented with one executer. The second is implemented with two executers and so on up to five executers. Each time 50, 100, 150, 200, 250, 300 pages are executed for parsing and indexing. The execution time for each run is observed. Figure (8) shows the effect of using more computers, as execution nodes, on time and the number of crawled pages. These experiments show improvement in the execution time of the indexing process using distributed indexing [5].

4-3 Searching Evaluation :

Information retrieval systems are usually compared on the basis of the "quality" of the retrieved document sets. This "quality" is traditionally quantified using two metrics, Recall (R) and Precision (P) [20]. Recall and Precision can be defined as:

$$R= r / K \quad …….(1) \qquad P = r / N \quad ……..(2)$$

Where : r = Number of relevant Web-pages retrieved.

N = total number of Web-pages retrieved.

K = Total number of web-pages that in the answer key.

To measure recall over a collection, it is required to mark every document in the collection as either relevant or non-relevant for each query in a list test of queries. This, of course, is an impossible task for WWW which contains billions of documents. Researchers have addressed this problem by developing standard document collections with queries and associated relevance judgments, and by limiting the domain of documents that are judged [21] which is not available for Arabic language. So, evaluation of the system will be based on precision measure only (P). Changes on the Recall measure (R) will be traced in different searching modes by considering the changes in the numbers of the retrieved and related Web pages for each input query.

Table (1) shows a comparing values for precision and the number of the retrieved/related Web pages for different input queries in different searching modes.

From Table (1) it is noted that the number of retrieved/related pages has increased (Enhances in Recall) due to the use of the synonym and the morphological modules with positive or negative changes in precision. It is noted also that using morphological search or/and synonyms search improves the average precession.

5 CONCLUSION

This research demonstrates an experimental prototype Intelligent Arabic Internet Search Engine which is based on the grid computing and Artificial Intelligence. The crawling and indexing loads are distributed over a set of computers, Using the grid computing techniques and tools, to increase the efficiency. Also the grid computing techniques and tools have been used partly during the searching module. The proposed architectures of the crawler and the indexer support the freshness feature through the dynamic indexing sub-module. The distribution of the index on many computers considering alphabetic order contributes in enhancing the search efficiency. Experiments showed that using the grid computing techniques and tools improves the crawling and indexing processes' efficiencies in particular when the number of computers in the grid increases. Experiments showed, also, that using Artificial Intelligence techniques like Arabic Morphological analyzer/generator and Synonyms have very positive effects on both Precision and Recall measures where the average precision exceeds the value of 0.92. This design is implemented for Arabic language but it suits any other language with language-related modifications.

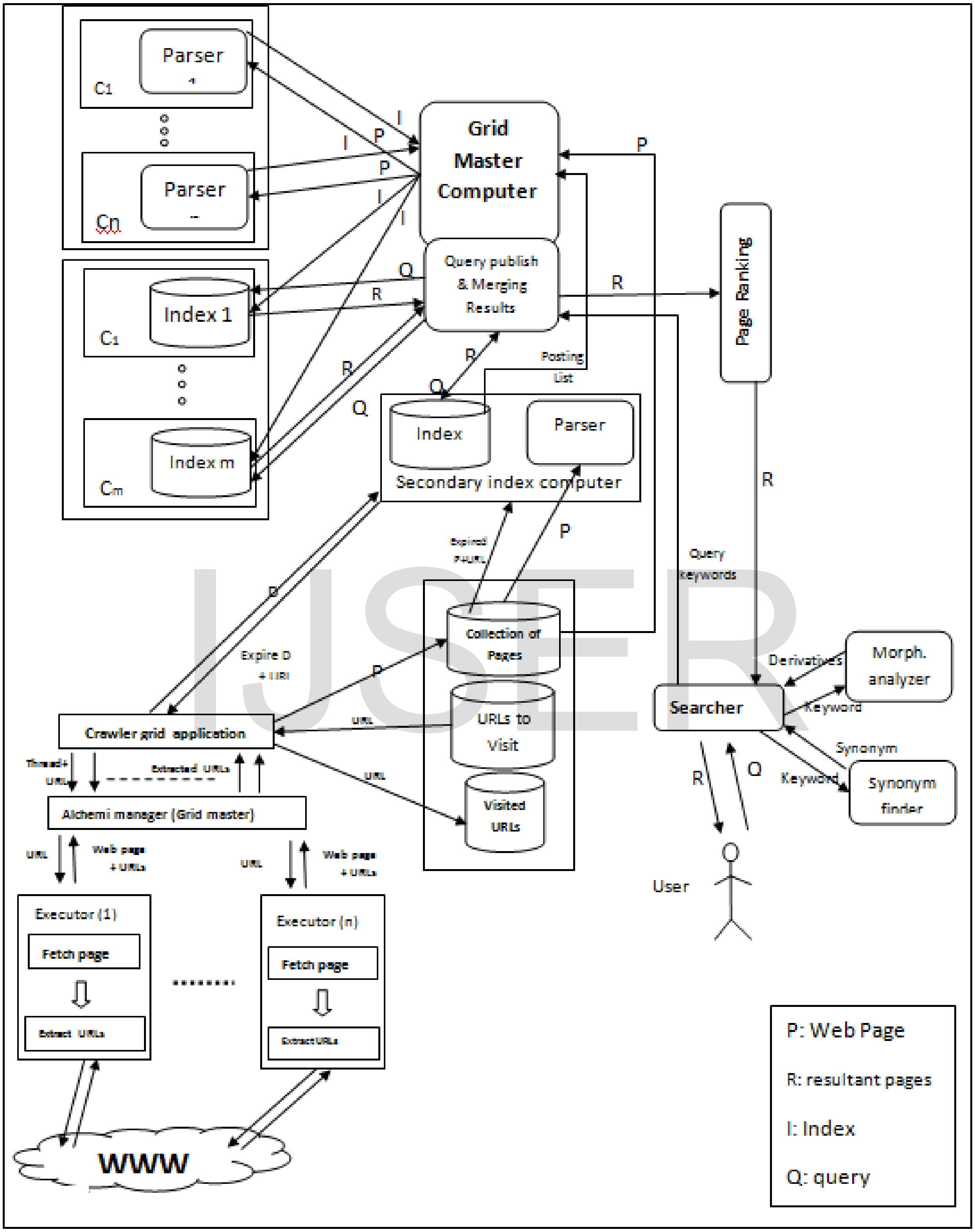

Figure (6): The proposed search engine components architecture.

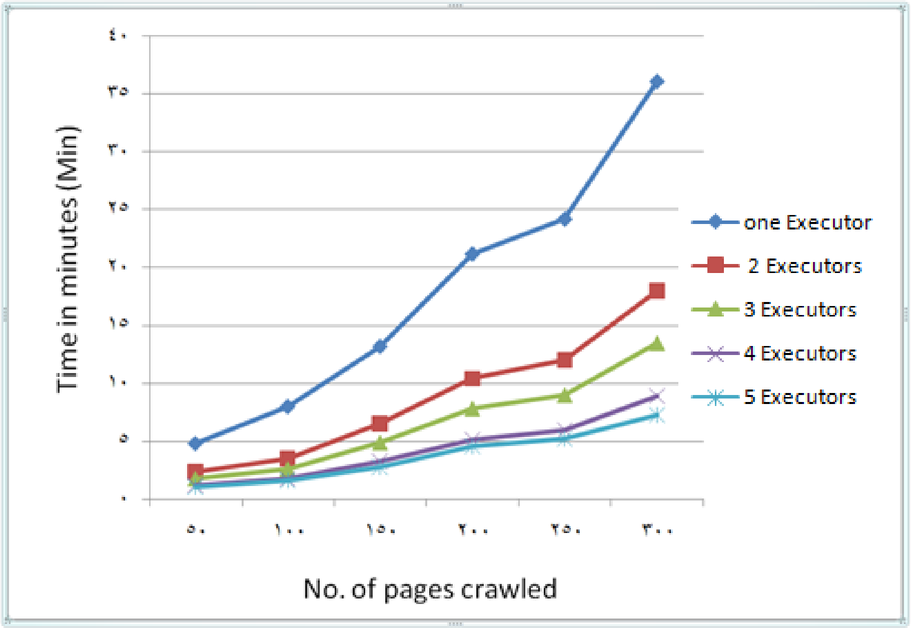

Figure (7) Number of crawled pages-time relation with different number of executers.

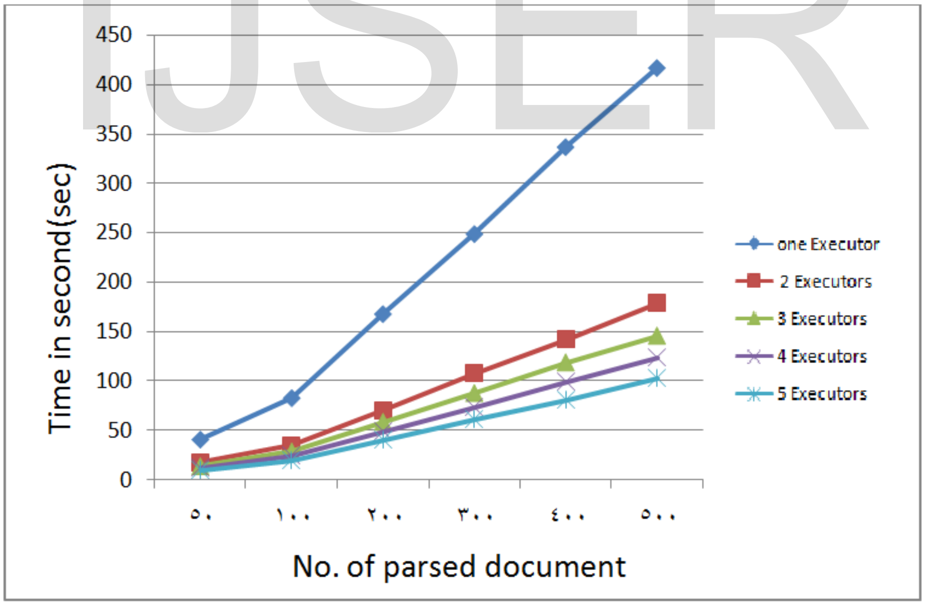

Figure (8): Number of indexed pages and time relation with different number of executers

Table (1): Precision and the number of the retrieved/related Web pages for different input queries in different searching modes.

	NORMAL SEARCH			SYNONYM SEARCH			MORPHOLOGICAL SEARCH			COMBINED SEARCH		
Queries	Numb.of Rec. Pages	Numb.of Related Pages	Precision	Numb.of Rec. Pages	Numb.of Related Pages	Precision	Numb.of Rec. Pages	Numb.of Related Page	Precision	Numb. of Rec. Page	Numb.of Related Page	Precision
Query 1	5	🟥 5	1	7	🟥 7	1	41	🟥 39	0.975	43	🟥 41	0.953
Query 2	11	9	0.818	11	9	0.818	21	18	0.857	21	18	0.857
Query 3	12	🟥 10	0.833	60	🟥 58	0.966	19	🟥 17	0.894	79	🟥 75	0.949
Query 4	0	0	0	0	0	0	31	31	1	31	31	1
Query 5	7	7	1	17	15	0.882	0	0	0	17	15	0.882
Average precision			0.7302			0.7332			0.7452			0.9282

REFERENCES

[1] Foster I., C. Kesselman, J. M. Nick, S. Tuecke, “The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration”, Global Grid Forum, June 22, 2002.

[2] Berman F., G. Fox, and T. Hey, Book: “Grid Computing: Making the Global Infrastructure a Reality”, published March 2003.

[3] Sakre M. et al, " An Arabic internet search engine supported with A.I. features ", International Journal of Intelligent Computing and Information sciences, Vol. 8, No.1, January 2008, Ain Shams University, Cairo.

[4] Sakre M. et al, “Crawler Architecture using Grid Computing”, International Journal of Computer Science & Information Technology (IJCSIT) Vol. 4, No. 3, June 2012.

[5] Sakre M. et al, “Dynamic and Distributed Indexing Architecture in Search Engine using Grid Computing”, International Journal of Computer Applications (IJCA) Vol. 4, No. 3, Oct 2012.

[6] C. Castillo, "Effective Web Crawling", Ph.D. Thesis, Dept. of Computer Science - University of Chile, November 2004.

[7] Ian Foster, Carl Kesselman, Steven Tuecke. " The anatomy of the grid: Enabling scalable virtual organizations", International Journal of High Performance Computing Applications Fall 2001 15: 200-222.

[8]. B. Barla Cambazoglu, Evren Karaca, Tayfun Kucukyilmaz, Ata Turk, Cevdet Aykanat, " Architecture of a grid-enabled Web search engine", Information Processing and Management (Elsevier), Vol. 43, pp: 609–623, 2007.

[9] Ahuja, Laxmi., " Use of Grid Computing in Search Engine - A Survey ", International Journal of Advanced Research in Computer Science , Volume 3, No.6,Nov.2012(Special Issue)

[10] Rajleen Kaur et al, " A Review Paper on Evolution of Cloud Computing, its Approaches and Comparison with Grid Computing " , (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (5) , 2014, 6060-6063.

[11] Rajshri S. Patill et al., " Data Replication & Caching in Data Grid :A Review ", (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 6 (1) , 2015, 459-462.

[12] Fareed, N.S. et al., "Enhanced semantic arabic Question Answering system based on Khoja stemmer and AWN ", Computer Engineering (ICENCO), 2013 9th International Conference, IEEE xplore digital library.

[13] Glenn Haya, Frank Scholze and Jens Vigen, "Developing a Grid-Based Search and Categorization Tool", High Energy Physics Libraries Webzine, issue 8, October 2003.

[14] SAS Global Forum 2011, Designing a Grid Computing Architecture: A Case Study of Green Computing Implementation Using SAS®, Krishnadas N, Indian Institute of Management, Kozhikode, Kerala, India, Paper 366-2011.

[15] Gulli A., A. Signorini. , "The Indexable Web is More than 11.5 billion pages", ACM 1595930515/05/0005, Chiba, Japan , May 10–14, 2005.

[16] Ramez el Masri, , shawkant B.Navathe. , “ Fundamentals of Database systems “, Pearson International Edition, 5th edition, 2007.

[17] Sakre M. et al, " Arabic Internet search engine supported with AI techniques ", The International Journal of Intelligent Computing and Information Sciences, Ain Shams Univeristy, Cairo, Egypt. January 2008.

[18] Ibrahim M. M., “ Information Retrieval of Arabic text using A.I. Techniques”, M.Sc. thesis, Military Technical College, Cairo, Egypt, 1987.

[19] Gerald Gazder and Chris Mellish, “ Natural language processing in prolog “, Addison – Wesley Publisher Company, 1989.

[20] Chowdhury, G. “ Introduction to modern information retrieval”, second edition, Facet publishing, 2004.

[21] E. Voorhees, “Overview of the English Text Retrieval Conference (TERC-8), Proceedings of TERC-8, 1999.

[22] Akshay Luther, Rajkumar Buyya, Rajiv Ranjan & Srikumar Venugopal, " Alchemi: A .NET-based Grid Computing Framework and its Integration into Global Grids", Technical Report, GRIDS-TR-2003-8, Grid Computing and Distributed Systems Laboratory, University of Melbourne, Australia.

Dr. Mohammed Mahmoud Ibrahim Sakre

Assoc. Professor of Computer Science,

Ex- Vice Dean for Academic and Students affairs,

Ex-The head of the Management Information Systems Department,

High Institute of Computers & Information Technology,

Shorouk Academy,

Shorouk City, Cairo, EGYPT.

E-mail: m_sakre2001@sha.edu.eg, m_sakre2001@yahoo.com