World Applied Sciences Journal 23 (8): 1053-1060, 2013
ISSN 1818-4952

DOI: 10.5829/idosi.wasj.2013.23.08.970

A Virtualized Information Indexing and Filtering Method for Web Contents Reside on Remotely Communicated Networks

Ijaz Ali Shoukat, Abdullah Al-Dhelaan and Mohsin Iftikhar
Computer Science Department, College of Computer and Information Sciences,
King Saud University, Riyadh, Saudi Arabia

Submitted: May 7, 2013; **Accepted:** Jun 15, 2013; **Published:** Jul 22, 2013

Abstract: The utilization of clouds in accessing of remote resources has evolved tremendous advantages like parallel execution, virtualization and supremacy of processing. Rather to filter queried information through single remote server is many times better to process it from a centralized cloud. Present architecture of information indexing and filtering mechanism is far away from cloud computing technology, however, their storage is already following the distributed concepts. In this paper, we developed a virtual indexing and cloud based information filtering model that follows cloud computing concept in *query filtering*, *virtual image of information indexes* and *searching* of indexed contents. We calculated architectural complexity and indexing probability of prior searching metrics in order to build cloud based information filtering and virtualized indexing model. We used mathematical comparison and practical query based method to evaluate the *indexing* and *filtering cost* of our proposed model. We analyzed that utilizing of cloud concepts with distributed storage at the time of filtering queried information is more effective rather to prior information filtering metrics.

Key words: Virtualization · Cloud Networks · Information filtering · Information Retrieval · Search Engines

INTRODUCTION

The quantity of web pages and storage of remote contents have been increased up to large scale due to the rapid growth of World Wide networks. This situation is seriously insisting the implementation of cloud computing technology in crawling of web. Persistent web storage provides inadequate coverage of web contents. The index coverage of search engines are only limited 40% to 50% as agreed by Les Beckford and Joe DeCicco [17]. In order to enhance the indexing mechanism, there is need of large storage and extra computational powers for bulky indexing coverage and searching ease. Moreover, additional processing power is also required to search large array of information index. Therefore, according to our opinion there is need to establish a virtualized image of indexing after filtering it in cloud like fashion. On the other hand there is no need to store virtual indexing and no need to host the original files on the clouds.

Virtual utilizations of remote components are the preferred mechanism today. The "*IBMSmartCloud*" fully adoptes the virtualization concepts for building variety of cloud services like public clouds, private clouds and hybrid clouds [1]. On the other hand the traditional primary crawlers are free from virtualized cloud environments and maintain their own indexing database. Firstly, user queries to web server and web server transfers the query to indexing server and eventually the index server forwards the query to document server of parent party where the file is actually hosted. Consequently the document server returns the results to the user. These kinds of search engines are fast and more popular so called primary robot, crawler, spider or automatic first tier search engines as depicted in Figure 1. Meta crawlers do not maintain their own indexing database but these are dependent on the primary crawling based spiders to seek documents with capability to collect more results by following the below architecture as shown in Figure 2. Hybrid crawlers are the advanced form of Meta crawlers which follow combined approach of primary search engines plus Meta search engines which means they also maintain their own indexing with ability to fetch indexing list from primary crawler based spiders. The Architecture of Hybrid Crawlers is shown in Figure 3.

Corresponding Author: Ijaz Ali Shoukat, Computer Science Department, College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia.

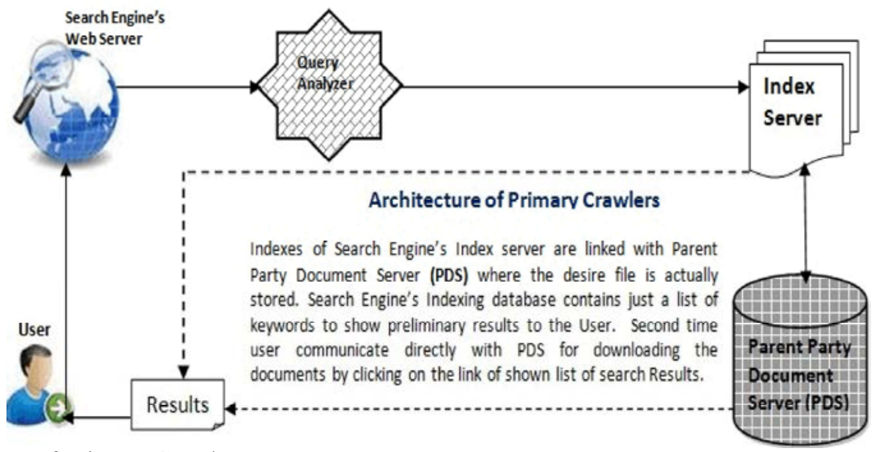

Fig. 1: Architecture of Primary Crawlers

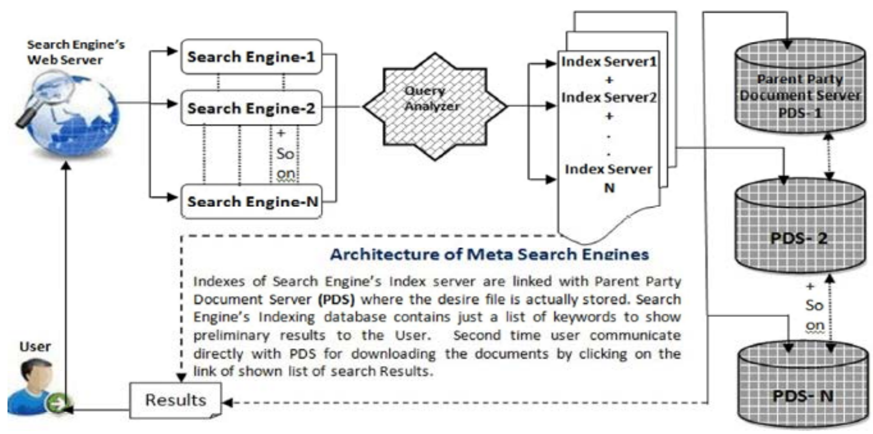

Fig. 2: Architecture of Meta Crawlers

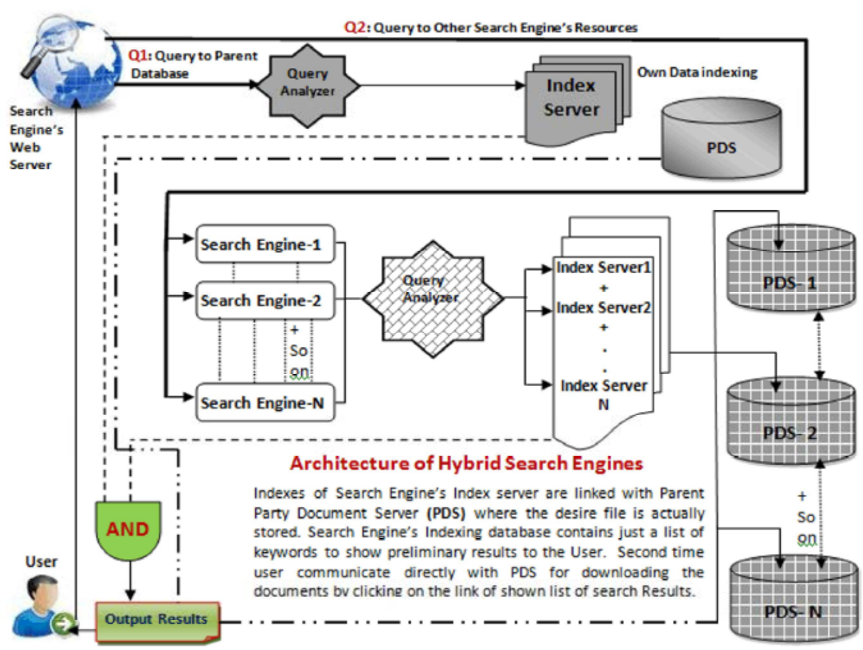

Fig. 3: Architecture of Hybrid Search Engines

The results provided by these search engines are large in capacity but they take more time than primary crawlers e.g. Mamma search engine. Human Power based Scholar Directories are the storage directories maintained by the scholars or highly skilled persons to index and store research oriented articles. Furthermore, there are some other kinds of directories which are used to store specific informative files with subscription or without subscription. Moreover, informative Paid or Free Inclusions are those kinds of directories, which are openly available to submit informative documents and presentations with subscription or without subscription. Rather to single search engine the combinations of two or more searching crawlers can cover more information with a capability of producing more results.

Recent trends have evolved cloud computing concepts in searching of information. The authors of study [2] proposed information retrieval method that can rank query with different percentage of matching file but their method is only limited to ranking of informative queries and finding the co-relation among the user queries in cloud environment. Our scheme differs with prior work and proposes virtualization of resources in cloud environments. This paper provides comparison among all types of web crawlers with respect architectural characteristics, searching cost, technology, quality of searching results and financial matters. Finally this article concludes a virtualized information indexing and filtering method to offer optimal searching for researchers, scientists, teachers, students, business professionals and even for home users with regardless of their study and specialized field. Moreover, it is significant for further research to develop future information indexing and filtering mechanisms to achieve optimal efficiency with affordable cost.

Literature Review: The elastic availability of remote resources is the significant trait of virtualized cloud environments. According to the authors of study [3] virtualization concept is beneficial for software developer to utilize available resources with greater efficiency. Despite of this, preliminary effort to introduce non-virtualized and cloud free environment based searching infrastructure was initiated in 1993 [4]. In November 2010 a study [5] introduced a hybrid clustering algorithm for acquiring of information. Another considerable issue regarding the WWW is that it does not follow a specific standard to maintain information movement and deletion which causes the hurdles of broken or dead links [6]. For better achievement of searching results, Lexical analysis of eight frequencies has been experimentally implemented on Altavista, MSN, Google and Yahoo in which combined hybrid methodology of "GoogleYahoo" remained better than the other two for retrieving more results [7]. Sabine Schneider and Max Stricker [8] have reported several information retrieval techniques such as Collaborative Filtering, Content-Based Filtering, Demographic Personalization, Utility-Based Information Retrieval, Knowledge-Based Recommendation, Hybrid Methods, Conversational Recommendation, Personalized Query Expansion and Context-Depend Methods but they prefer to utilize personalized information methodology under any type of leading crawlers. In order to crawl the web in limited time the study [9] has presented hybrid architecture by utilizing the pipelining concept of parallelism and which made possible to get domain specific information with hybrid crawler in limited time span. To find out resemblance between searching and recommendation a study [10] has been conducted in 2005. In this paper a hybrid technique called RankFeed has been put forwarded by implementing three kinds of approaches [11] (1) Content Approach - it means document appearing sets are similar to previous contents (2) Usage Approach - it means mostly visited documents are recommend for good quality (3) Web Page Mining - it means the combined integration of two approaches. According to study [11] it has been concluded that IRF ranking is better than "PageRank" Method for single web site. A survey of information acquiring and filtering has been published in 1995 reported major techniques for information scanning like signature files, text mapping and inversion and clustering [12].

According to the authors of study [13] the optimal performance and efficiency of any information filtering method is greatly concern with its server response, up time, mean time between failure (MTBF) and mean time between repair (MTBR). An other study [14] recently claimed that Google crawler is best in response time, uptime, MTBF and MTTR rather to Mamma, Yahoo, Gigablast, MSN and Excite. The study [15] reported in 2011, that the indexing size of Google is more than 25 billion of web pages. Therefore, due this reason we selected Google indexing server in our proposed model, however, the reason to choose CiteSeerX is its quality of contents without any subscription cost. Our approach aims to achieve virtualized searching environment by utilizing the available resources for information retrieval and indexing point of view. This approach is significantly cost effective and efficient to operate. Present literature fully supports this approach. The authors of study [16] reported that virtualization technology not only cost effective but it also optimize the performance of running applications.

MATERIALS AND METHODS

For proposing an ideal cloud based virtualized searching method, we firstly found the architectural characteristics (pros and cons) of each type of traditional un-virtualized web crawlers. After that we computed the searching cost and architectural complexity by using quantitative approach. Finally to analyze the proposed model practically, we selected five types of queries in which some are specific and some are more generalized form to judge the maximized enhanced capability of showing authentic results upon desired source of search. This way creates a virtual list of contents filter in cloud environments. Against each query we analyzed the top 10 results of Google, Google Scholar and CiteSeerX to map authentic information like PDF files. For those cases when there is no PDF result(s) shown in top 10 results of each search engine then we considered HTML results that contain authentic information. We made the comparison of our proposed scheme with prior searching tools through practical experimentations in order to conclude, how our proposed cloud based model is able to produce more quality of results in limited time and searching effort under affordable architecture complexity.

Architectural Characteristics: Some architecture related characteristics of various traditional web crawlers are compared in a Table 1.

Table 1: Characteristics Comparison of Different Crawlers

Searching Methods	Advantages	Disadvantages
Primary Crawlers or Spiders	• Contain their own indexing server. • Single source of data searching • Provide efficient searching with affordable time. • Provide automated caching and indexing. • Mostly documents are freely accessible. • Best in case of introductory or basic overview related details	• Result's quality is marginal. • Sometime provide results with dead links. • Provide many of un-necessary results. • Some results are un-authentic with basic informative details without references.
Meta Searching Methods	• Their searching is based on two or more data seeking sources • Provide vast range of Searching results. • Mostly documents are freely accessible. • Best in case of introductory or basic overview related details • Some time provide results with some outdated or dead links. • Mostly results are un-authentic with basic informative details without references.	• Searching requires more penalty of time than primary crawlers • They have not their own indexing database. • Vast range or un-necessary results. • No quality control.
Hybrid Searching Methods	• Maintain their own indexing database by using combined Meta methods. • Their un-necessary range of results is less than meta method's searching range. • It is good approach to get many results with less time than the Meta method. • Mostly documents are free accessible	• Searching time penalty is greater than primary crawlers. • Sometime provide results with dead links. • Provide many of un-necessary results rather than the primary crawlers. • Quality control depends on hybrid services. • Some results are un-authentic with basic informative details without references.
Human Power Directories	•Maintain their own database of documents. • Documents are powered by experts scholars like editors, authors and reviewers. • Provide no un-necessary results • Provide quality of results. • Searching time is less than primary crawlers and Meta search engines.	• Mostly the document's accessing need privileges or membership cost.
Informative Paid or Free Inclusions	• Mostly they are freely available for any kind of information or advertisement. • Some are requires fee to store information. • Contain their own database of documents. • Searching time is not longer but depends on the usage of services.	• Some are requires fee to store information. • Not recommended approach. • Provide un-necessary or limited results • No quality control. • Mostly they do not contain scholar documents. • Sometime information is not complete. • Mostly provide un-authentic results. • Not too much popular.

Table 2: Complexity and Cost Comparison

Search Engines	General Searching Methods	Database & Indexing	Architecture Complexity	Architectural Searching Time Cost without Parallelism
Google	Primary Crawling Searching Method	Personal indexing Database	Low	UQ→WS→QA→InX→U→DB(FHS)→U Search Time Cost = 6 ms. UQ : User query to WS : Web Server QA : Query Analyzer InX : Index Server DB : Document Database U : User FHS : File hosting Server (Parent Party)
Yahoo	Primary Crawling Method	Personal indexing Database	Low	UQ→WS→QA→InX→U→DB(FHS)→U Search Time Cost = 6 ms.
MSN	Primary Crawling Method	Personal indexing Database	Low	UQ→WS→QA→InX→U→DB(FHS)→U Search Time Cost = 6 ms.
Gigablast				UQ→WS→QA→InX→U→DB(FHS)→U Search Time Cost = 6 ms.
Mamma	Hybrid Method	Personal + Other(s)	High	UQ→WS→QA→N(Servers)→N(InXs)→U→ FHS→U Cost = Cost of Arrows + N time cost (Servers +InXs) + β(X) Cost = { 7 + K * Σ(NTC) + β(X) } ms K= Servers + Indexing DBsΣ(NTC): n time cost of servers and indexing DBs
Excite	Meta Methods	No personal indexing and Database as it lends services from others	High	UQ→WS→QA→N(Servers)→N(InXs)→U→ FHS→U Cost = Cost of Arrows + N time cost (Servers +InXs) + β(X) Cost = { 7 + K * Σ(NTC) + β(X) } ms K= Servers + Indexing DBs Σ(NTC): n time cost of servers and indexing DBs
Altavista	Meta Methods	No personal indexing and Database as it lends services from others		UQ→WS→QA→N(Servers)→N(InXs)→U→ FHS→U Cost = Cost of Arrows + N time cost (Servers +InXs) + β(X) Cost = { 7 + K * Σ(NTC) + β(X) } ms K= Servers + Indexing DBs Σ(NTC): n time cost of servers and indexing DBs
AOL		No personal indexing and Database as it lends services from others	High	UQ→WS→QA→N(Servers)→N(InXs)→U→ FHS→U Cost = Cost of Arrows + N time cost (Servers +InXs) + β(X) Cost = { 7 + K*Σ(NTC) + β(X) } ms K= Servers + Indexing DBs Σ(NTC): n time cost of servers and indexing DBs
Ask	Primary Crawling Method	Personal indexing Database	Low	UQ→WS→QA→InX→U→DB(FHS)→U Search Time Cost = 6 ms.
AlltheWeb	Meta Method	No personal indexing and Database as it lends services from others	High	UQ→WS→QA→N(Servers)→N(InXs)→U→ FHS→U Cost = Cost of Arrows + N time cost (Servers + InXs) + β(X) Cost = { 7 + K * Σ(NTC) + β(X) } ms K = Servers + Indexing DBs Σ(NTC) : n time cost of servers and indexing DBs
AliWeb	Primary Crawling Method	Personal indexing Database	Low	UQ→WS→QA→InX→U→DB(FHS)→U Search Time Cost = 6 ms.
Lycos	Meta Method	No personal indexing and Database as it lends services from others	High	UQ→WS→QA→N(Servers)→N(InXs)→U→ FHS→U Cost = Cost of Arrows + N time cost (Servers + InXs) + β(X) Cost = { 7 + K * Σ(NTC) + β(X) } ms K = Servers + Indexing DBs Σ(NTC): n time cost of servers and indexing DBs
Dogpile	Meta Method	No personal indexing and Database as it lends services from others	High	UQ→WS→QA→N(Servers)→N(InXs)→U→ FHS→U Cost = Cost of Arrows + N time cost (Servers +InXs) + β(X) Cost = { 7 + K * Σ(NTC) + β(X) } ms K = Servers + Indexing DBs Σ(NTC): n time cost of servers and indexing DBs
CiteSeerX Directory and Search Engine	Directory in the form of Scientific Library and Search Engine	Personal indexing Database	Low	UQ→WS→QA→InX→U→DB(FHS)→U Search Time Cost = 6 ms.

All searching methods have been analyzed and discussed under their affirmative and feeble characteristics as summarized in Table 1. Primary searching methods have low complexity and marginal searching scope but their searching time is minimum. On the other hand Meta and hybrid methods provide vast range of results but their searching time is greater than primary crawlers.

Architectural Complexity and Searching Cost Comparison: For measuring the searching cost and complexity, we analyzed how actually query is processed according to the defined architecture of any web crawler. For each step, we showed an arrow and assumed its cost (1 millisecond) to compute complexity and searching cost of leading search engines as discussed in a Table 2.

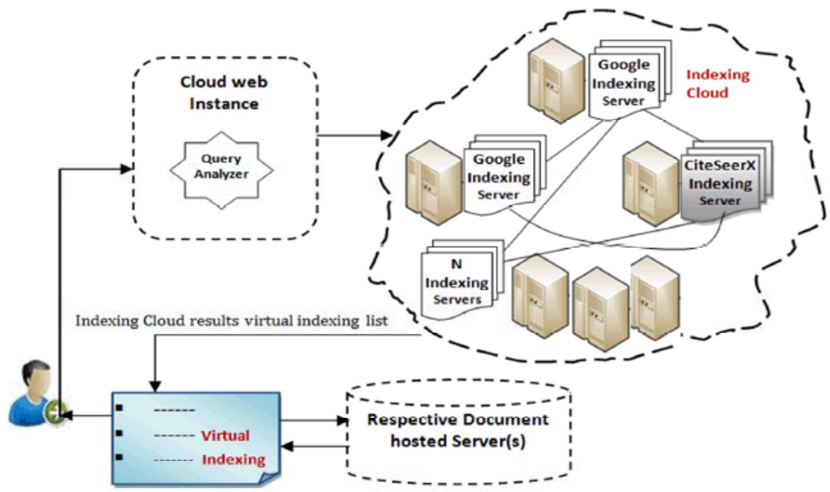

Fig. 4: Virtual Information Indexing and Retrieval Cloud

Proposed Cloud Base Information Indexing and Retrieval Model: After analyzing and comparing theoretical and mathematical soundness of architectural characteristics, complexity and searching cost estimation of various types of traditional searching crawlers; we proposed a cloud based virtual approach of Google + CiteSeerX to get virtualized indexes of information as discussed in a Figure 4. User can get their desired contents by clicking it from virtualized indexing list. For minimizing the architectural complexity and searching cost of our proposed virtualized cloud Model, we implemented a parallelism technique at the time of issuing and processing of user query which reduces its searching time and make this model as an efficient virtualized approach.

The Working Flow of Our Approach is as Follows: The user queried to cloud web instance that transfers the user query to clouds. In cloud query is executed parallel through joint fashion in individual environment of each remote resource that is the actual trait of creating cloud. Finally the cloud returns the virtual index list back to the user from where user can click and download his/her file from parent party file hosting server. The working model of our proposed virtualized information indexing and retrieval model is depicted in Figure 4. Mathematically, the working flow of our proposed scheme can be written as follows:-

$$UQ \rightarrow Cloudweblinstance \rightarrow Cloud \\ \left(N - Index\ Servers\right) \rightarrow U \rightarrow FHS \rightarrow U \quad (1)$$

Architectural Complexity and Searching Cost Comparison: We compared the architecture complexity and searching cost of our proposed scheme with the traditional hybrid and Meta filtering schemes. Information filtered under hybrid or meta scheme takes {7 + K * Σ(NTC) + β(X) } ms to filter the required query but our proposed model takes {5 + K * Σ(NTC) + β(X) } ms filtering cost as follows:-

Hybrid Crawlers Architecture searching Cost: Cloud base Proposed Virtualized Architecture searching cost

- UQ→WS→QA→N(Servers)→N(InXs)→U→ FHS→U
- Cost = Cost of Arrows + N time cost (Servers +InXs) + β(X)
- Cost = { 7 + K * Σ(NTC) + β(X) } ms

UQ	:	User query to
WS	:	Web Server (Searching)
QA	:	Query Analyzer
InX	:	Index Server
DB	:	Document Database
U	:	User
FHS	:	File hosting server (Parent Party)
β(X)	:	Small factor of parallel processing
Note	:	We suppose each arrow has 1 ms cost

- K = Servers + Indexing DBs
- Σ(NTC) : n time cost of indexing servers
- UQ→Cloud Web Instance→Cloud (N-Index Servers)→U→FHS→U
- Cost = Cost of Arrows + N time cost (cloud Indexing) + β(X)
- Cost = { 5 + K*Σ(NTC) + β(X) } ms

Table 3: Practical Search Results

Selected Queries	Google Result probability relationship	Google Scholar Result Probability relationship	(Google + Cite SeerX) Result Probability relationship
EEG based Brain Computer Interface	R̈₱(G) >R̈₱(GS)	R̈₱(GS) < R̈₱(G)	Σ(GX) → 2 times R̈₱(G)
Cache memory issues	●₱ (G) >R̈₱(GS)	R̈₱(GS) <R̈₱(G)	Σ(GX) → 2 times R̈₱(G)
performance evaluation of bit torrent	R̈₱ (G) >R̈₱(GS)	R̈₱(GS) < R̈₱(G)	Σ(GX) → 2 times R̈₱(G)
linear searching algorithm	R̈₱ (G) >R̈₱ (GS)	R̈₱(GS) <R̈₱(G)	Σ(GX) → 2 times R̈₱(G)
Review article	R̈₱ (G) > R̈₱ (GS)	R̈₱(GS) <R̈₱(G)	Σ(GX) → 2 times R̈₱(G)

G = Google
GS = Google Scholar
GX = CiteSeerX + Google
R̈₱ = Result Probability
Σ(GX) = Total results of (G + GX)
∈→ = belongs to or equivalent to

Equivalency Comparison:
Prior Scheme = Proposed Scheme

$$7+K*\sum(NTC)+\beta(X) = 5+K*\sum(NTC)+\beta(X)$$
$$7 = 5ms.$$

Hence, Prior scheme cost is 7 ms and proposed scheme cost is 5 ms.

This cost comparison clearly invokes that the searching and architectural complexity of proposed scheme is outperformed to prior filtering schemes. Primary crawlers takes 6 ms and hybrid or meta crawler takes {7 + K * Σ\(NTC) + β(X) } ms that is larger as compared to our proposed virtualized indexing and information retrieval model.

Practical Experimental Results: In order to test the proposed virtualized cloud model in practical point of view that how our model is robust incase of generating larger and quality oriented virtual contents of information, we used query method to test the indexing capacity with content's quality on traditional remote applications like Google, Google Scholar and after that we used the same heuristic on our proposed combination of (Google + CiteSeerX beta version) in order to compared the practical results. The probability of showing valid authentic results acquired through different searching methods are summarized in Table 3. The term *valid authentic* result means published PDF or HTML contents that are available freely with full text to users. For comparative analysis, we selected only top 10 results of each search engine because mostly users focus on first page of searching results. Consequently we found the following probabilistic relationship from experimental observations as summarized in Table 3.

In top 10 searching results when any search engine fails to show any PDF result then we considered HTML valid authentic result's probability otherwise we only considered the probability of PDF results for analysis. On the other hand Google Scholar shows more number of authentic information rather than Google but many of its results require either *full text file charges* or membership login details which are based on membership charges of those platforms where file is actually hosted. So many users do not have membership and mostly user want to get quality of results in free of cost. In case of getting authentic results without any charges Google is observed better than Google Scholar as reported in Table 4. The probabilistic results (Table 4) claim that there is measurable difference among the results of all three sources and proposed virtualized information indexing and filtering cloud is superior to the existing ones.

CONCLUSION

Virtualization of remote resources in cloud based environment is outperformed and cost effective rather to traditional metrics of questing information via single remote source. Furthermore, the quality of contents, minimum querying effort, least searching time and low architectural complexity is the priority of end users that can effectively be fulfilled through virtualization and cloud computing concepts. Our experimental judgments (Table 4) and mathematical cost comparison (section 3.4) clearly exhibit that our proposed cloud based virtual information indexing and filtering model is cost effective as well as 2 times outperformed than prior information filtering methods. Moreover, our proposed model is superior to provide quality of information with limited searching effort, affordable architectural complexity as compared to the traditional information filtering schemes.

ACKNOWLEDGEMENT

The authors extend their appreciation to the Deanship of Scientific Research at King Saud University for funding this work through research group no. RGP-VPP-264.

REFERENCES

1. Marinescu, D.C., 2013. Cloud Computing: Theory and Practice, 1st. Edition, Pages 416. ISBN: 9780124046276, Elsevier Publisher,
2. Liu, Q., C.C. Tan, J. Wu and G. Wang, 2012. Efficient Information Retrieval for Ranked Queries in Cost-Effective Cloud Environments, The 31st Annual IEEE Conf. on Comp. Communication, pp: (2581-2585), 978-1-4673-0775-8/12 IEEE, 2012.
3. Johnsen, E.B., R. Schlatte and S.L.T. Tarifa, 2012. “Modeling Resource-Aware Virtualized Applications for the Cloud in Real-Time ABS”, Accepted in Proc. Intl. Conf. on Formal Engineering Methods (ICFEM 2012), LNCS,
4. Dujmoviæ, J. and H. Bai, 2006. Evaluation and Comparison of Search Engines Using the LSP Method, Comput. Sci. Inform. Syst., 3: 31-56.
5. Banu, W.A. and P.S. Abdul Kader, 2010. “A Hybrid Context based Approach for Web Information Retrieval”, International Journal of Computer Applications, 10(7): 0975-8887.
6. Lawrence, S., D.M. Pennocet, G.W. Flake, R. Krovertz and F.M. Coetzee, 2001. Persistence of Web Reference in Scientific Research, in Proceedings of IEEE 34:(2), pp: 26-31.
7. Park, S.T. and D.M. Pennoc, 2002. Analysis of Lexical Signatures for Finding Lost or Related Documents, SIGIR’02, August pp: 11-15. Tampere, Finland,
8. Schneider, S. and M. Stricker, 2010. “Personalized Information Retrieval and Web Search”, Tech. Rep. Graz University of Technology,
9. Das1, A. and S. Nandy, 2010. “ Hybrid Focused Crawler - A Fast Retrieval Of Topic Related Web Resource For Domain Specific Searching”, International Journal of Information Technology and Knowledge Management, 2(2): 355-360.
10. Kiewra, M., 2005. “RankFeed - Recommendation as Searching without Queries: New Hybrid Method of Recommendation”, Journal of Universal Comp. Sc., 11(2): 229-249.
11. Kazienko, P. and M. Kiewra, 2003. “ROSA - Multi-agent System for Web Services Personalization”. In Proc. First Atlantic Web Intelligence Conference Proceedings, LNAI 2663, Springer Verlag, 2003b, pp: 297-306.
12. Faloutsos, C. and D.W. Orad, 1995. “A Survey of Information Retrieval and Filtering Method”, funded by the National Science Foundation under Grants IRI - 9205273 and IRI-8958546 (PYI),
13. Shoukat, I.A. and M. Iftikhar, 2011. “Performance Analysis of 5A’s Web Robots”, Scientific Research and Essays, 6(12): 2564-2577.
14. Shoukat, I.A., W.S. Ansari, M. Iftikhar, J.Taheri and A.Y. Zomaya, 2012. “Practical Assessment of Remotely Communicated End-Point Servers of Leading Web Crawlers”, in Porc. of 9th IEEE Joint Conference on computer Science and Software Engineering, pp: 219-224, Thiland,
15. Shoukat, I.A., M. Iftikha and A. Haseeb, 2011. “Indexing Size Approximation of WWW", International Journal of Advanced Research in Computer Science, 2(2): 71-75.
16. Sabahi, F., 2012. “Secure Virtualization for Cloud Environment Using Hypervisor-based Technology”, International Journal of Machine Learning and Computing, 2(1): 39-45.
17. Beckford, L., J. DeCicco and T. Lam, 2000. Search Engines, Technical Report, POC: Stephen Parshley,