Towards Improving Exploratory Faceted Search Results

Mohammed Najah Mahdi[1], Abdul Rahim Ahmad[2], Roslan Ismail[3]

[1]Institute of Informatics and Computing in Energy, College of Computing & Informatics (CCI), Universiti Tenaga Nasional, Jalan Ikram-Uniten, 43000 Kajang, Selangor, Malaysia, najah.mahdi@uniten.edu.my

[2]Department of System & Networking, College of Computing & Informatics (CCI), Universiti Tenaga Nasional, Jalan Ikram-Uniten, 43000 Kajang, Selangor, Malaysia, abdrahim@uniten.edu.my

[3]Department of Software Engineering, College of Computing & Informatics (CCI), Universiti Tenaga Nasional, Jalan Ikram-Uniten, 43000 Kajang, Selangor, Malaysia, roslan@uniten.edu.my

ABSTRACT

Finding information on the web, containing billions of documents, is becoming a serious issue. Search engines (SE) help users find handy information such as videos, text or pictures. For open-ended results, users need to perform queries in order to find enough information about their topic of interest. This can be difficult when huge numbers of relevant and irrelevant results are returned from the queries causing information overload. Certain SEs incorporating visualization allow easy understanding of the documents and often reduce the Information overload issues. Sorting of more relevant results from the less relevant ones is difficult since the former is insufficient. This is obvious when the Web grows and the information therein increases to become larger data repositories. The query and response (lookup) processes are often used by most current SEs. Exploratory Search (ES) can provide open ended search. Therefore, the overall search can be improved by applying the Faceted search (FS), which is one of the features of ES. Studies have mentioned that the overloaded data can be reduced when traditional SEs are replaced with the ES, through their features. The prototype design we are working on is based on the idea of ES engine that can be used to visualize the information when exploring the Web. This concept is aimed at supporting more active means of finding information through coordinated visualizations and reducing large amount of search efforts required per query. The prototype system was evaluated for performance and compared with CloudMining SE. The results show that our prototype gives average improvement of 20% relevant results when compared with CloudMining SE.

Key words: Exploratory Search, Faceted Search, Visualization, Search Engine.

1. INTRODUCTION

The Web has become a universal repository of human culture and knowledge in history. Considering millions of Internet users, creating hundreds of billions of documents, finding information on the web, requiring submission of queries to search engines, has become a serious challenge [1].

The most popular search tool on the Web is the search engines (SEs), one of the purpose is to help researchers find handy information: text, pictures, or videos [2]. When users type a few words, the engine returns a list of information relevant to their queries. Accessing information, through the SE, can be easier and faster on the Web.

Today, vast amounts of online information can be searched through SE using sophisticated means. Thus, the SEs are considered by many people as their primary means to seek information [3]. In April 2016, Google, which is capable of processing over 50,000 queries per second, received 5.5 billion searches per day approximately, while the rate of search per year was close to 1.2 trillion searches worldwide. Meanwhile, Yahoo! and Bing received 3,200 and 927 searches per second, respectively [4]. These huge numbers of searches show how important the SEs are and how people depend on them [5]. With the advent of the Internet, most IT departments support more storage systems possessing computational power available for Internet applications such as SE [6]. Other examples include the cloud computing through which vast information can be accessed. In this study, a SE prototype system, based on a cloud computing platform, is presented, through which the idea of a visual ES engine is developed for information exploration on the Web. This concept is aimed at supporting a more active means of finding information through coordinated visualizations by reducing the large amount of SE effort needed per query, which are frequently used by commercial search engines. This paper is organized as follows: the related work is presented in section II, through which the interactive visual and guides on ES and FS for SEs are reviewed. Section III gives the overview of SE framework with some illustrations of the methodology, through which the data were collected and the results were implemented. In section IV, the paper discusses the test and evaluation of the proposed prototype search system. Finally, the conclusion section summarized on how the relevant search results on Web can be improved.

2. RELATED WORK

The scope of finding information on the Web is large; this makes the search process difficult to users with a lot of

returned search results. For example, in May 2016, Web was estimated to have more than 3 billion documents and most researchers check only the first few results of their searches [7, 8].

Considering the dynamic variation of Web, the traditional SEs lack the quality and intelligence to judge the nature of documents invoked by the users. This is because most existing SEs only rank the documents according to their futures [9]. The insufficient number of related results was investigated through the effects of ambiguity query [10]. The hypothesis states that less ambiguous queries return more relevant results. This further indicates that a method which is relevant to the structure of the desired document is simply the robust method of sorting the result.

A. Exploratory Search

The information-seeking problem context which are open-ended, persistent, and multi-faceted can be described by ES. The ES, in the first sense, is often applied on learning, scientific exploration and decision making contexts. In the second sense, the ES strategy can be applied to all techniques of seeking information in such a way that the preference and experiences of the seeker can be reflected as much as possible [11, 12].

A number of SE activities differentiating ES from lookup search have been identified by Marchionini [11]. ES is basically relevant to learn and investigate activities. The activities are classified, by Marchionini, into three major types (see Figure 1). When the information provided by the users is well defined, look up can be sufficient to discover the searched result. However, when the information provided by the users which describes the users' need is ill defined, the lookup search can only be necessary; not sufficient for the users seeking information for either learning or investigation.
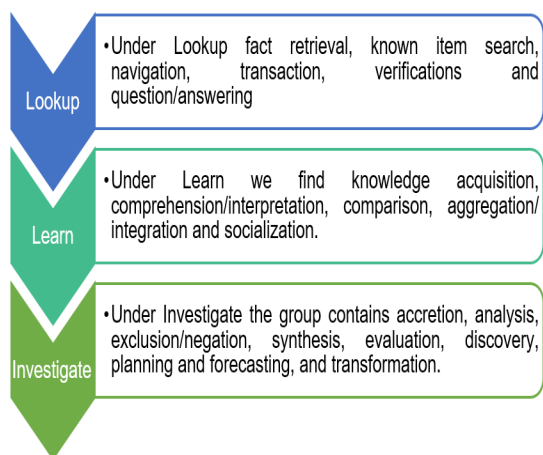

Figure1: Exploratory Search Classification Activities

At all stages of the information-seeking process, the ES illustrates a shift from the analytic platform of query-information matching toward direct guidance.

B. Faceted Search Systems

Faceted search (FS) has grown to be well-developed and popular as a search technique for business and educational websites[13, 14]. Because of the enormous information on the Web, a searcher looks for information provider that can help them analyze such information and provide valuable output that will be helpful to their research[15]. FS is a novel exploratory search engine process that can help researcher discover relevant information within a large document repository. This give researchers the opportunity to explore a set of documents by utilizing a number of discrete attributes or facets[16].

3. THE FRAMEWORK

The summary of a SE framework is presented in Figure 2. Two sides are involved here; the client and cloud server. While the former allow the users to visually analyze and determine within the relation graphs when particular queries are used, the latter returns relationship graphs through metasearch and knowledge integration[17]. Three major components are composed in the framework interpretation and delegation of visualization; (1) relationship graphs on the Web via Similarity Search, (2) visual ES of relationship graphs through both querying and browsing strategies and (3) human-computer interactions via the multi-interface. The following subsection describes the design process key design guideline and some details of aesthetic design.
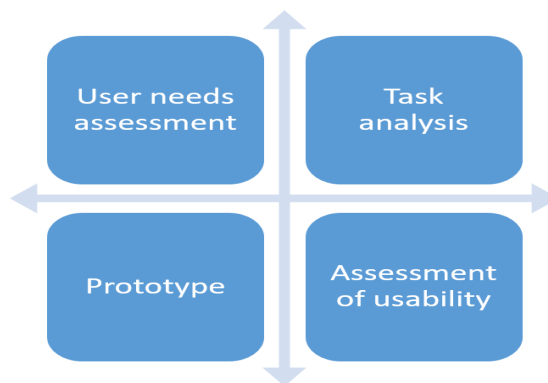

Figure 2: User-Centered Design Approach

A. Designing Process

The web interfaces always displace a user-centered approach in layout. As mentioned in Figure 2, a series of steps are involved in the process, through which users are constantly solicited[18].

B. User Interaction

The examples assembled with prototype all share the ssimilar user type cooperation disregarding of the information sort being examined. The aspects of search, facets, visualization and query by instance are all considered. To make the interface, a conventional faceted search interface is adopted and extended with exploratory capabilities[21]. Not only the

interface, but also the overall sought user interaction can be focused on. The particulars of the interface and its customization are discussed in the next parts. Fig. 3 display the series of steps taken for designing the process in detail:
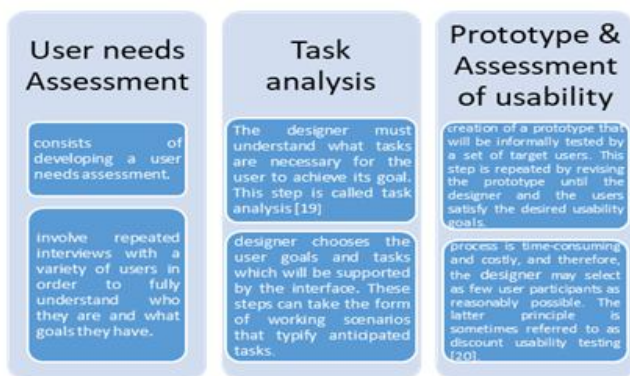

Figure 3: Designing Steps

C. Software Modeling

From a software perspective, the functionalities of the prototype modules must be decoupled from each other as much as possible. We preferredd to form the prototype a web application, which requires different elements for its various ES tasks. Each module is software that can be used separately. For instance, fSphinx module is used for FS[22], whereas Similarity Search module is used as an item based SE which implements Bayesian Sets[23]. It performs item-based search, and it can be distributed to scale into extensive datasets of tens of millions or even billions of items.

D. Exploratory Visual Search

The important feature of an ES system is itscapabilityy to demonstrate data in a way that favors emergence arrangements and provides new insights. Various types of visualizations can be employed on search outcomes or facets[24].

The system supports three different forms of visualizations on facets only. The searched outputs are so far not yet visualized. The following example demonstrates how the tag cloud view can be used to identify new information about the document corpus more simply than by only looking at the counts of the facet rates [25]. From a back-end aspect, an instance is quietly a web application that provides the whole articulate with a level of customizability. The following section discusses how facets can be displayed within the interface[26].

E. Facet Visualization

Facets, aside from their refining competence, give an impressive arbitrary of the searched output with according to facet arrangement. They can reveal patterns of distribution and occurrences at an cumulative level discussed ways to present facet values graphically [15]. Selecting the right

visualization can constantly shift the focus from finding to other ES tasks, including data analysis. Three different types of views, a tag cloud, a histogram count, and a more exotic rose diagram, are provided. Nevertheless, new views can simplybe developed and added. Using facet views in system is best characterized with the consecutive example[26].

The general plan objective was not to withdraw a lot from ordinary FS practices. Thusly, the interface ought to furnish the capacity to blend things in with customary faceted metadata determinations. The outcomes ought to be a lot of comparative things limited to the aspect selection(s) or potentially full content hunt. The aspect gathering capacity ought to be adjusted to utilize the similitude search scores. For instance, the terms in a label cloud could be weighted by the closeness scores notwithstanding its recurrence. The interface ought to likewise give input regarding why the archives have coordinated. This is significant so as to assist clients with shaping a psychological model about the basic coordinating calculation.

4. EVALUATIONS AND VALIDATIONS

Several systems were introduced and implemented, recently. These systems gained some success in various information overload and ambiguous query. However, those systems have some limitations, particularly when present information and the speed of the system are considered.

A. Evaluation on the Performance of Benchmarked

The Exploratory Faceted Search Systems by Ksikes [27] on the systems has gained some success to build a system dealing with problem Information Overload. This was used as a benchmark for proposed Search Systems because the prototype main features work on exploratory search system (ESS) by Ksikes [27].
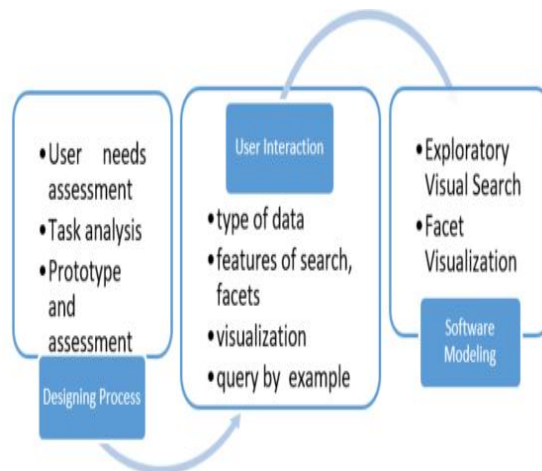

Figure 4: Overview of the Search Engine Framework

The Cloud Mining proposed by Ksikes [27] used three large publicly available datasets: the Instance Building of Cloud Mining. The first step deals with downloading the data of interest and the size of the different instances meant to

download potentially millions of documents from the web. The relevant information also had to be extracted and loaded into a database. The second step consists of setting up the back-end. However, the Weaknesses of Cloud Mining features are envisioned in the short term and long term as described below.

First would be to add search views as pluggable widgets. These could include a map view and a space-time view; Second, CloudMining could feature different skins, for different devices such as tablets or phones. Third, an entity extraction module should be provided. Fourth is the lack of metadata issue. Five, each instance could make use of a social module to vote, comment, edit or curate the documents found. This is a currently important missing piece of CloudMining which should help in implementing the pipelining idea approach to information overload, Six, similarity search should be improved in order to allow for queries made of items proposed in the information collection.

5. RESULTS AND DISCUSSION

We conducted some experiments with the SE we have developed. The goal of these experiments is to evaluate how our proposed SE is performing. We presented some example queries that a user may have. We tested those queries as input to the system and presented the output. Then we carefully discussed each of those results and discover if they are accurate or not. In the following sections, we discuss more about our experiments.

A. A Performance Evaluation Based Search Engines

The evaluation and analysis of performance is often conducted on the newly designed SEs. A clear example includes the work of Khan, et al. [28] who conducted evaluation to ascertain the capability of Semantic web based SEs, as well as the Keyword based SE. This was achieved by selecting three Semantic web based SEs (Lexxe, DuckDuckGo and Bing) and two queries based SEs (Google and Yahoo), through which their performances were compared. Ten queries were taken from various topics and ran on these SEs, through which their precision ratios for these queries were calculated. Their work was based on proposing a framework used for semantic based SE that can resolve the problem of synonyms and polysemy terms. These used ontologies based crawler in place of web crawler for effective precision.

Other examples include Kim, et al. [29], whose work prepared 40 queries when assuming a scenario through which a user search for a movie with partial information spanning over many elements. Some words, such as the title, the name of an actor, or even the year the movie was produced, may be remembered. These are all combined into a query, as summarized in Table I. By manual searching, a relevant document could be obtained since most queries were designed to find a single movie.

Table I: Query Lists

Query	Query	Description
Q1	Dog Hachi	Movie with title " Hachi: A Dog's Tale" , directed by LasseHallström
Q2	FBI agent crime	" Catch Me If You Can " Frank Abagnale Jr. (book)
Q3	Sylvester boxer	A sport movie featuring Sylvester Stallone
Q4	Assassinate Robot	" The Terminator ", a crime movie
Q5	Wizard return home to Kansas	" The Wizard of Oz " directed by Victor Fleming

The initial experiment by Kim (2009), along with the IMDB dataset, consisting of 437,281 documents or XML records was studied. Each document, constructed from text data, corresponds to a movie. The element types include the "title", "year", "release data", "language", "genre", "country", "location", "colour info", "cast", "team" and "plot".

Following this, the search used in our experiments, was conducted based on study by Kim, et al. [30], to evaluate the performance of searching of two SEs (our prototype vs Cloud Mining). The comparisons were based on the meaning of the keywords, as well as their relationships between the keywords. The web pages and the number of relevant document retrieved were also compared. Five queries representing a broad range of IMDb queries adapted from Kim, et al. [30] were used, each of which was submitted to the SEs. First ten documents retrieved, each classified as "relevant" or "irrelevant", were evaluated using binary human relevance judgment. Following this, the Precision to Recall ratio of keyword based SE were calculated for five documents. In order to avoid temporal variations by retrieval of relevant data from each SEs, each query was executed in two different SEs at nearly the same time. The criteria for the evaluation of retrieved documents were defined into 5 categories as shown in Table II. The evaluation is on the basis of the following criteria; the number of related information retrieved by each SE for the first ten information retrieved of prototype, and CloudMining.

B. Relevant Documents Retriever for CloudMining SEs

In searching for the 5 queries, not all queries can return results. The total sites that returned results were 127. Out of that, only 14 sites returned results were selected and compared with that in the study by Kim. Majority of the selected query results were from query 5 because it returned the most. Table II illustrates the relevant statistics for the 5 queries and the selected sites.

Table II: The Search Engine Results Criteria

No	Category	Criteria
1	Relevant	If the content of the web page closely matched the subject matter of the search query, then it was categorized as "relevant" and it was given a score of 1.
2	Same query, but different web addresses relevant	If the content of the web page is not closely related to the subject matter, but consists of some relevant aspects to the subject matter of the search query, then it was categorized as "relevant" and it was given a score of 1
3	Non-relevant	If the content of the web page is not related to the subject matter of the search query, then it was categorized as "Non-relevant" and it was given a score of 0
4	Duplicate document	If the content of the web page consisted of a whole series of Duplicate document, also define non-relevant, it was given a score of 0.

The table below shows that the results of query with respect to the search, evaluate the performance for different queries (query Q1 to query Q5) that can be analysed and summarized.
Some analysis is given below:

a) For queries (Q1and Q2), their results are shown in SERP returned. The total number of selected sites was 2. Meanwhile, the relevant documents retrieved for selected queries were 1 and the partial results were also 1. b) The results for both queries Q3 and Q4 are showed in SERP returned. The relevant (Category 1) sites are 2 for Q3 and 122 for Q4. The SERP showing irrelevant sites (Category 3) was the highest because they are matching 1 for Q3 and no matching for Q4 with queries. c) The results for query (Q5) are shown in SERP returned. The irrelevant site (Category 3) is 1. No results showing any comparisons.

The best performance was found in Q4. The number of related information retrieved by CloudMining for the information retrieved based on the features was used to reduce the user's effort in terms of relevant sites. However, a problem is shown with metadata keyword, when the user conducts a deep search in the repository layer by layer.

This result illustrates that CloudMining performance in keywords and the polysemy words needs to be improved. Also, the growth of the datasets shows a negative effect in the performance. CloudMining performance needs to be improved in large datasets, as well. In addition, it needs to improve the user learning in term of capability performance of the system. This is shown in Table III.

Table III: Number of Relevant Documents Retriever for CloudMining

Queries	Total no. Site	Selected Site	1	2	3	4
Q1	2	2	1	1	0	0
Q2	2	2	1	1	0	0
Q3	2	1	1	0	0	0
Q4	122	10	1	0	9	0
Q5	1	1	0	0	1	0
Total	127	14	4	2	10	0

C. Relevant Documents Retriever for our Prototype SEs

In the search using the 5 queries, most queries returned results. The total sites that returned results were 138. Out of it, only a total of 17 sites that returned results were compared with the study by Kim. Majority of the selected query results were from query 1 because it returned the most. Table IV illustrates the relevant statistics for the 5 queries and the selected sites. It is clear from the table that 9 of the sites were related (category 1) and 4 of sites were less (category 2). Non-relevant (category 3) was 3, while the duplicate document (category 4) was 0.

Table IV shows the outputs of query with respect to the SE precision for the different queries (query Q2 to query Q5) that can be analyzed and summarized. The prototype has better results than CloudMining as shown in Table VI. This is because the prototype adopted additional features and item based search to help get more accurate matching and more similarity results. Some analysis is given below: a) The results for the queries Q3 and Q4 are shown in SERP returned. The total number of sites were 120; the selected sites were 4 for Q4. The results showed a higher matching with relevant sites (Category 1). The same query had different relevant results (Category 2) and non-relevant (Category 3). The queries showed a higher matching and similarities with the selected sites. The relevant documents retrieved were 1 of 2 for Q3. The matching results that were retrieved for all relevant sites that are in IMDb sets. b) For queries (Q2 and Q5), all the queries are selected, the most relevant sites (Category 1) are 2. The SERP has shown that prototype had the ability to search and find all the related sites using the selected keyword. c) For query Q1, the results of the queries as shown in SERP returned. The total number of sites were 8, the selected sites were 4. The matching results retrieved all relevant sites that 3 of 4 were saved in IMDb sets.

This result illustrates that our prototype's performance is better in keywords as well as polysemy words. Also, the growth of the datasets showed a positive effect in the performance. The prototype has improved in the learning and guidance for the users in term of enhancement in the capability performance of the system.

Table IV: Number of Relevant Documents Retriever for prototype SEs

Queries	Total no. Site	Selected Site	1	2	3	4
Q1	8	4	3	0	1	0
Q2	5	5	2	1	1	1
Q3	2	1	1	0	0	0
Q4	120	4	1	2	1	0
Q5	8	4	2	1	1	0
Total	138	17	8	4	4	1

The number of related information retrieved by each SE for the first ten documents retrieved is presented in Table VI. It

shows the percentage of the average improvement of our prototype in terms of the five cases of keyword compared to the other approaches using different keyword values with respect to the active situation in IMDb. The improvement percentage is achieved by the proposed approach listed with appropriate different keyword values. For example, the prototype obtains an average improvement of 20% in term of recall. The average improvement is calculated for all of the values of queries. In this paper we have shown the evaluation of SE performance like Semantic SE and Keyword Based SE. In our experiment the prototype retrieves more related documents when compared with CloudMining. With regard to overall performance, the prototype retrieved more relevant documents when compared with the entire selected SE. This reduces the problem of synonyms and polysemy words. Meanwhile, our experimental output showed that SE performance of prototype SE was higher than the Keyword SE.

The overall graphical precision ratio of SE for first ten documents has been presented (see Fig. 5). It is clear from the figure that prototype retrieved more relevant documents than CloudMining SE, which accounts for 43% of the original query. Meanwhile, the Precision Ratios of IMDb, CloudMining and prototype is shown in Fig 4. These results were plotted into a Search Queries and precision graph. The graph was interpolated at 5 points to show the precision value of the retrieval performance capabilities at each Query point in the search results. Fig. 5 summarizes the effectiveness of all systems, as well the algorithms used per graph. Item Based search and IMDb outperformed the other algorithms; CloudMining algorithms followed by prototype. The systems had precision values started from 0, and then slightly increased. There were no relevant data for the selected query, or they had a different result not related to the selected query. The prototype had precision values that start at 0 and slightly increase at every query point. The query point had a more relevant data. This makes our prototype higher than the other systems. For example, at query Q4, the figure showed a high precision that reached 160 for the selected query as well as more relevant data retrieved by our prototype.
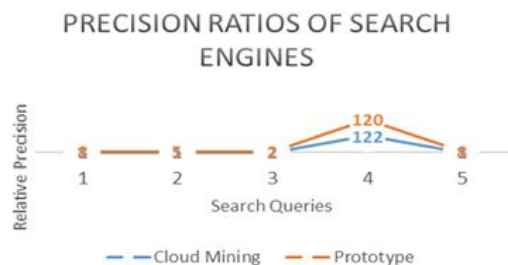

Figure 5: Precision Ratios of Search Engines for First Ten Documents

D. Conclusion on the Performance Evaluation Based SEs

We have shown the performance comparison of CloudMining engines and our prototype search engine. Using our experiment, the prototype system retrieved more relevant information when compared to that of CloudMining. This comparison was based on their overall performance. Among the advantages of this search engine is how it would reduce the problem of partial information spanning over many elements even if the user doesn't have the right words to write the query. Our experimental output showed that SE performance of the prototype SE was higher than all the selected SEs.

6. CONCLUSION

What can be drawn from this research is related to the issue of insufficient number of related outputs. These were obtained by investigating the effect of query ambiguity on the search outputs. It can be seen that queries that are less ambiguous return more relevant outputs. The insufficient sorting of the results was addressed by investigating three methods of sorting the outputs.

In recent years, among the great interests in Web searches is how the relevant search results can be improved. The Web is heterogenous in nature: large volumes, and varied structures. Thus, finding compatible outputs according to the needs of every individual is challenging.

Interactive graphical and visualization techniques are accordingly suggested to increase the ability of the display to handle large numbers of outputs. Meanwhile, they should simultaneously present several attributes of each Web page. In addition, query reformulation and reconstruction can usually be controlled by the SE.

REFERENCES

[1] A. Figueroa and G. Neumann, "Context-aware semantic classification of search queries for browsing community question‑answering archives," Knowledge-Based Systems, vol. 96, pp. 1-13, 2016.

[2] R. Qumsiyeh and Y.-K. Ng, "Searching web documents using a summarization approach," International Journal of Web Information Systems, vol. 12, pp. 83-101, 2016.

[3] J. P. Ribeiro and J. P. Peixoto, "Searching for Events through Social Media: An Exploratory Study," International Journal of Marketing, Communication and New Media ISSN, pp. 2182-9306, 2014.

[4] C. Smith, "100 Amazing Google Statistics and Facts," 2016.

[5] R. Sentance, "Search Engine Watch," 2016.

[6] J. Bates, P. Best, J. McQuilkin, and B. Taylor, "Will Web Search Engines Replace Bibliographic Databases in the Systematic Identification of Research?," The

Journal of Academic Librarianship, vol. 43, pp. 8-17, 2017.

[7] M. R. Baye, B. De los Santos, and M. R. Wildenbeest, "Search Engine Optimization: What Drives Organic Traffic to Retail Sites?," Journal of Economics & Management Strategy, vol. 25, pp. 6-31, 2016.

[8] J. Bughin, "Big data, Big bang?," Journal of Big Data, vol. 3, p. 1, 2016.

[9] N. Ibrahim, A. H. Chaibi, and H. B. Ghézala, "Scientometric re-ranking approach to improve search results," Procedia Computer Science, vol. 112, pp. 447-456, 2017.

[10] I. Ounis, C. Macdonald, and R. L. T. Santos, "Search result diversification," Foundations and Trends in Information Retrieval, vol. 9, pp. 1-90, 2015.

[11] G. Marchionini, "Exploratory search: from finding to understanding," Communications of the ACM, vol. 49, pp. 41-46, 2006.

[12] R. W. White, G. Marchionini, and G. Muresan, "Evaluating exploratory search systems: Introduction to special topic issue of information processing and management," ed: Pergamon, 2008.

[13] A. C. Robinson and S. D. Quinn, "A brute force method for spatially-enhanced multivariate facet analysis," Computers, Environment and Urban Systems, vol. 69, pp. 28-38, 2018.

[14] M. N. Mahdi, A. R. Ahmad, and R. Ismail, "Paradigm Extension of Faceted Search Techniques A Review," Journal of Telecommunication, Electronic and Computer Engineering (JTEC), vol. 9, pp. 149-153, 2017.

[15] M. A. Qureshi and D. Greene, "An Investigation into Information Navigation via Diverse Keyword-based Facets," in 24th Irish Conference on Artificial Intelligence and Cognitive Science (AICS'16), University College Dublin, Dublin, Ireland, 20-21 September 2016, 2016.

[16] Y. Tzitzikas, N. Manolis, and P. Papadakos, "Faceted exploration of RDF/S datasets: a survey," Journal of Intelligent Information Systems, vol. 48, pp. 329-364, 2017.

[17] R. Bhati and S. Prasad, "Open domain question answering system using cognitive computing," in Cloud System and Big Data Engineering (Confluence), 2016 6th International Conference, 2016, pp. 34-39.

[18] S. Ben and P. Catherine, "Designing the user interface,")^(Eds.):'Book Designing the user interface'(Reading, Mass.: Addison Wesley Longman, 1998, edn.), 2005.

[19] E. Goodman, M. Kuniavsky, and A. Moed, "Observing the user experience," Burlington, Massachusetts: Morgan Kaufmann, 2012.

[20] J. Nielsen, "Usability 101: Introduction to usability," ed, 2003.

[21] K. Athukorala, D. Głowacka, G. Jacucci, A. Oulasvirta, and J. Vreeken, "Is exploratory search different?A comparison of information search behavior for exploratory and lookup tasks," Journal of the Association for Information Science and Technology, vol. 67, pp. 2635-2651, 2016.

[22] A. Aksyonoff, Introduction to Search with Sphinx: From installation to relevance tuning: " O'Reilly Media, Inc.", 2011.

[23] J. Park and S.-H.Yook, "Bayesian Inference of Natural Rankings in Incomplete Competition Networks," Scientific Reports, vol. 4, p. 6212, 08/28/online 2014.

[24] B. Sarrafzadeh and E. Lank, "Improving Exploratory Search Experience through Hierarchical Knowledge Graphs," in Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2017, pp. 145-154.

[25] O. Hoeber and H. Liu, "Comparing tag clouds, term histograms, and term lists for enhancing personalized web search," in 2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology, 2010, pp. 309-313.

[26] E. Ali, "Dynamic Personalized Ranking of Facets for Exploratory Search," in Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2017, pp. 1379-1379.

[27] A. Ksikes, "Towards exploratory faceted search systems," University of Cambridge, 2014.

[28] J. A. Khan, D. Sangroha, M. Ahmad, and M. T. Rahman, "A performance evaluation of semantic based search engines and keyword based search engines," in Medical Imaging, m-Health and Emerging Communication Systems (MedCom), 2014 International Conference on, 2014, pp. 168-173.

[29] J. Kim, X. Xue, and W. B. Croft, "A probabilistic retrieval model for semistructured data," in European Conference on Information Retrieval, 2009, pp. 228-239.

[30] J. Kim, X. Xue, and W. B. Croft, "A Probabilistic Retrieval Model for Semistructured Data," in ECIR, 2009, pp. 228-239.

[31] Azham Hussain, S.V Manikanthan, Padmapriya T. and MahendranNagalingam. "Genetic algorithm based adaptive offloading for improving IoT device communication efficiency". Wireless Network, August, 2019. DOI: 10.1007/s11276-019-02121-4