NASA/TM—2016-219159

A Full-Text-Based Search Engine for Finding Highly Matched Documents Across Multiple Categories

Hung D. Nguyen and Gynelle C. Steele
Glenn Research Center, Cleveland, Ohio

September 2016

NASA STI Program . . . in Profile

Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA Scientific and Technical Information (STI) Program plays a key part in helping NASA maintain this important role.

The NASA STI Program operates under the auspices of the Agency Chief Information Officer. It collects, organizes, provides for archiving, and disseminates NASA's STI. The NASA STI Program provides access to the NASA Technical Report Server—Registered (NTRS Reg) and NASA Technical Report Server—Public (NTRS) thus providing one of the largest collections of aeronautical and space science STI in the world. Results are published in both non-NASA channels and by NASA in the NASA STI Report Series, which includes the following report types:

- TECHNICAL PUBLICATION. Reports of completed research or a major significant phase of research that present the results of NASA programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA counter-part of peer-reviewed formal professional papers, but has less stringent limitations on manuscript length and extent of graphic presentations.

- TECHNICAL MEMORANDUM. Scientific and technical findings that are preliminary or of specialized interest, e.g., "quick-release" reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.

- CONTRACTOR REPORT. Scientific and technical findings by NASA-sponsored contractors and grantees.

- CONFERENCE PUBLICATION. Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or co-sponsored by NASA.

- SPECIAL PUBLICATION. Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.

- TECHNICAL TRANSLATION. English-language translations of foreign scientific and technical material pertinent to NASA's mission.

For more information about the NASA STI program, see the following:

- Access the NASA STI program home page at http://www.sti.nasa.gov

- E-mail your question to help@sti.nasa.gov

- Fax your question to the NASA STI Information Desk at 757-864-6500

- Telephone the NASA STI Information Desk at 757-864-9658

- Write to:
  NASA STI Program
  Mail Stop 148
  NASA Langley Research Center
  Hampton, VA 23681-2199

NASA/TM—2016-219159

A Full-Text-Based Search Engine for Finding Highly Matched Documents Across Multiple Categories

Hung D. Nguyen and Gynelle C. Steele
Glenn Research Center, Cleveland, Ohio

National Aeronautics and
Space Administration

Glenn Research Center
Cleveland, Ohio 44135

September 2016

Level of Review: This material has been technically reviewed by technical management.

Available from

NASA STI Program
Mail Stop 148
NASA Langley Research Center
Hampton, VA 23681-2199

National Technical Information Service
5285 Port Royal Road
Springfield, VA 22161
703-605-6000

This report is available in electronic form at http://www.sti.nasa.gov/ and http://ntrs.nasa.gov/

Contents

A Full-Text-Based Search Engine for Finding Highly Matched Documents Across Multiple Categories

Hung D. Nguyen and Gynelle C. Steele
National Aeronautics and Space Administration
Glenn Research Center
Cleveland, Ohio 44135

Abstract

This report demonstrates the full-text-based search engine that works on any Web-based mobile application. The engine has the capability to search databases across multiple categories based on a user's queries and identify the most relevant or similar. The search results presented here were found using an Android (Google Co.) mobile device; however, it is also compatible with other mobile phones.

Introduction

Full-text-based search engines are the subject of continuous research and technological improvement (Refs. 1 and 2). These engines have two roles: indexing support and querying support. Most current open-source search engines are based on the full-text indexed techniques built in by MySQL (MySQL AB) database management systems. These database management systems feature a broad and robust set of document search capabilities (Refs. 3 to 5). Their query languages permit searching documents by using the "LIKE" operator, regular expressions (REGEX), and "FULLTEXT" indexes. Although they are very simple and easy to integrate into any application, engines have some limitations including (1) not being designed to handle large amounts of text data; (2) having relatively simple ranking mechanisms; (3) having virtually no control over the indexing; and (4) words must be at least four letters long. In addition, because these built-in full-text search algorithms are only supported by their own database storage engines, they are not compatible with each other. Many complex search engine algorithms have been developed by search engine companies, but the algorithms are often proprietary.

In general, the search function matches queries to create an index that lists words and their locations within each document. Thus, the function of an index algorithm determines whether a word exists within a particular document and is able to store information regarding the frequency of the word. The indexed results are then used to calculate, rank, and group such documents into corresponding categories. Document parsing, however, still presents many challenges when extracting necessary information for indexing.

Full-Text-Based Search Architecture

Figure 1 shows how a full-text-based search engine works. While incoming documents are converted into plain text using a document filter, an index module stores a list of essential words from each document into a database, then the database is optimized for quick lookups without storing the full text of each document. Once the user queries the system using a Web page, the Stoplist module deletes words that are not useful for the search. To find more relevant documents, the system will add all possible synonymous terms, which will provide more finely tuned search results. After the processed query is compared to the stored index, the weighting and ranking factors for each document can be computed across categories to determine the most relevant documents. When the user queries, the full text function accesses the optimized word index to identify which documents contain the requested terms. Records that match the query are returned, but records that do not match the query are not returned.

To demonstrate how the search function works, let us assume that there are three different categories—Category A1, Category A2, and Category A3, and under each category hundreds or thousands of documents could be stored as shown in Figure 2. While indexed documents are organized in the way that their contents are relevant to each category, they also store a list of words in a database for each document and the pointers to their locations. For this illustration, a number of M indexed documents are sorted under Category A1, N indexed documents are sorted under Category A2, and O indexed documents are sorted under Category A3, where M, N, and O can be any arbitrary number.

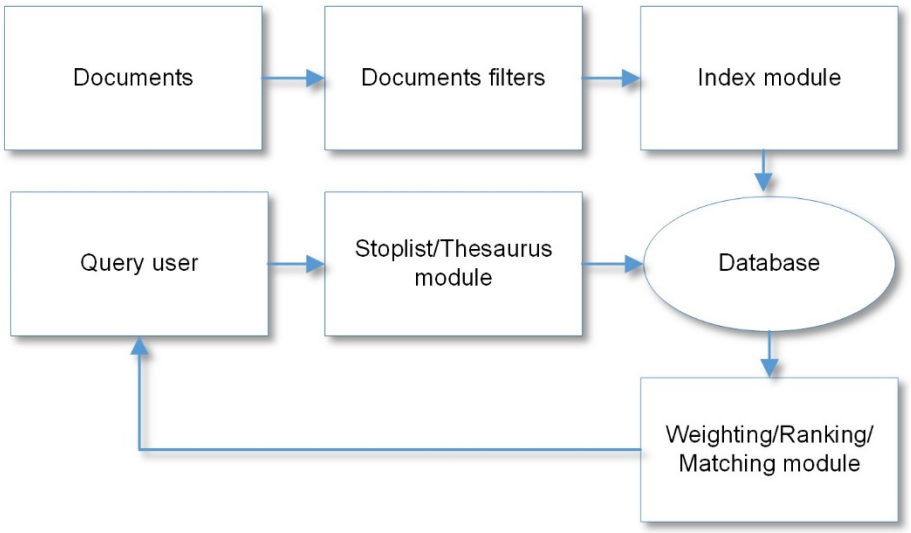

Figure 1.—Architecture of a full-text-based search engine.

Category A1	Category A2	Category A3
Indexed-Doc1-A1	Indexed-Doc1-A2	Indexed-Doc1-A3
Indexed-Doc2-A1	Indexed-Doc2-A2	Indexed-Doc2-A3
Indexed-Doc3-A1	Indexed-Doc3-A2	Indexed-Doc3-A3
Indexed-Doc4-A1	Indexed-Doc4-A2	Indexed-Doc4-A3
Indexed-Doc5-A1	Indexed-Doc5-A2	Indexed-Doc5-A3
Indexed-Doc6-A1	Indexed-Doc6-A2	Indexed-Doc6-A3
Indexed-Doc7-A1	Indexed-Doc7-A2	Indexed-Doc7-A3
Indexed-Doc8-A1	Indexed-Doc8-A2	Indexed-Doc8-A3
Indexed-Doc9-A1	Indexed-Doc9-A2	Indexed-Doc9-A3
Indexed-Doc10-A1	Indexed-Doc10-A2	Indexed-Doc10-A3
Indexed-Doc11-A1	Indexed-Doc11-A2	Indexed-Doc11-A3
Indexed-Doc12-A1	Indexed-Doc12-A2	Indexed-Doc12-A3
Indexed-Doc13-A1	Indexed-Doc13-A2	Indexed-Doc13-A3
Indexed-Doc14-A1	Indexed-Doc14-A2	Indexed-Doc14-A3
Indexed-Doc15-A1	Indexed-Doc15-A2	Indexed-Doc15-A3
*	Indexed-Doc16-A2	Indexed-Doc16-A3
*	Indexed-Doc17-A2	Indexed-Doc17-A3
*	Indexed-Doc18-A2	Indexed-Doc18-A3
Indexed-DocM-A1	Indexed-Doc19-A2	Indexed-Doc19-A3
	Indexed-Doc20-A2	
	Indexed-Doc21-A2	*
	*	*
	*	*
	*	Indexed-DocO-A3
	Indexed-DocN-A2	

Figure 2.—Indexed documents by category.

Search terms from the user's query are organized into the number of combinations shown in Figure 3 to identify elements in documents that could be indexed. Generally the formula for finding the number of combinations of "k" words to choose from a set of "n" words is

$$C_k^n = \frac{n!}{k!(n-k)!}$$

For example, if the four words (n=4) real-time, system, safety, and assurance are entered as search terms, by applying a formula above, one combination of four words and four combinations of three words (k=3) will result in a total of five groups as shown in Figure 3.

Four words and their synonymous terms are matched with all indexed documents associated with all categories. Documents that must contain all four words and some of the synonymous terms are assigned a weighting scale from 1 to 5 points, where 5 is the most relevant and 1 is the least relevant. The weighting scale is computed based on the frequency of those synonymous terms occurring in the indexed document. Documents that do not meet this requirement are assigned 0 points. The weighting computation result is shown in Figure 4.

Selection of the indexed documents can be achieved by mapping from the highest weighting scale to the lowest scale across all categories, and the result is stored in Class 1 as shown in Figures 5 and 6.

Group 1	real-time	system	safety	assurance
Group 2	real-time	system	safety	
Group 3	real-time	system	assurance	
Group 4	real-time	safety	assurance	
Group 5	system	safety	assurance	

Figure 3.—Groups of four-word queries.

Category A1	Weighting	Category A2	Weighting	Category A3	Weighting
Indexed-Doc1-A1	5	Indexed-Doc1-A2	4	Indexed-Doc1-A3	5
Indexed-Doc2-A1	5	Indexed-Doc2-A2	3	Indexed-Doc2-A3	1
Indexed-Doc3-A1	1	Indexed-Doc3-A2	5	Indexed-Doc3-A3	3
Indexed-Doc4-A1	3	Indexed-Doc4-A2	2	Indexed-Doc4-A3	0
Indexed-Doc5-A1	0	Indexed-Doc5-A2	1	Indexed-Doc5-A3	3
Indexed-Doc6-A1	3	Indexed-Doc6-A2	0	Indexed-Doc6-A3	3
Indexed-Doc7-A1	4	Indexed-Doc7-A2	0	Indexed-Doc7-A3	4
Indexed-Doc8-A1	5	Indexed-Doc8-A2	5	Indexed-Doc8-A3	0
Indexed-Doc9-A1	0	Indexed-Doc9-A2	3	Indexed-Doc9-A3	5
Indexed-Doc10-A1	0	Indexed-Doc10-A2	4	Indexed-Doc10-A3	3
Indexed-Doc11-A1	2	Indexed-Doc11-A2	0	Indexed-Doc11-A3	5
Indexed-Doc12-A1	2	Indexed-Doc12-A2	2	Indexed-Doc12-A3	1
Indexed-Doc13-A1	1	Indexed-Doc13-A2	1	Indexed-Doc13-A3	2
Indexed-Doc14-A1	4	Indexed-Doc14-A2	5	Indexed-Doc14-A3	3
Indexed-Doc15-A1	3	Indexed-Doc15-A2	2	Indexed-Doc15-A3	2
*		Indexed-Doc16-A2	4	Indexed-Doc16-A3	5
*		Indexed-Doc17-A2	3	Indexed-Doc17-A3	0
*		Indexed-Doc18-A2	2	Indexed-Doc18-A3	0
Indexed-DocM-A1	2	Indexed-Doc19-A2	0	Indexed-Doc19-A3	2
		Indexed-Doc20-A2	1		
		Indexed-Doc21-A2	4	*	
		*		*	
		*		*	
		*		Indexed-DocO-A3	0
		Indexed-DocN-A2	4		

Figure 4.—Weighting scale computation for real-time, system, safety, and assurance by category.

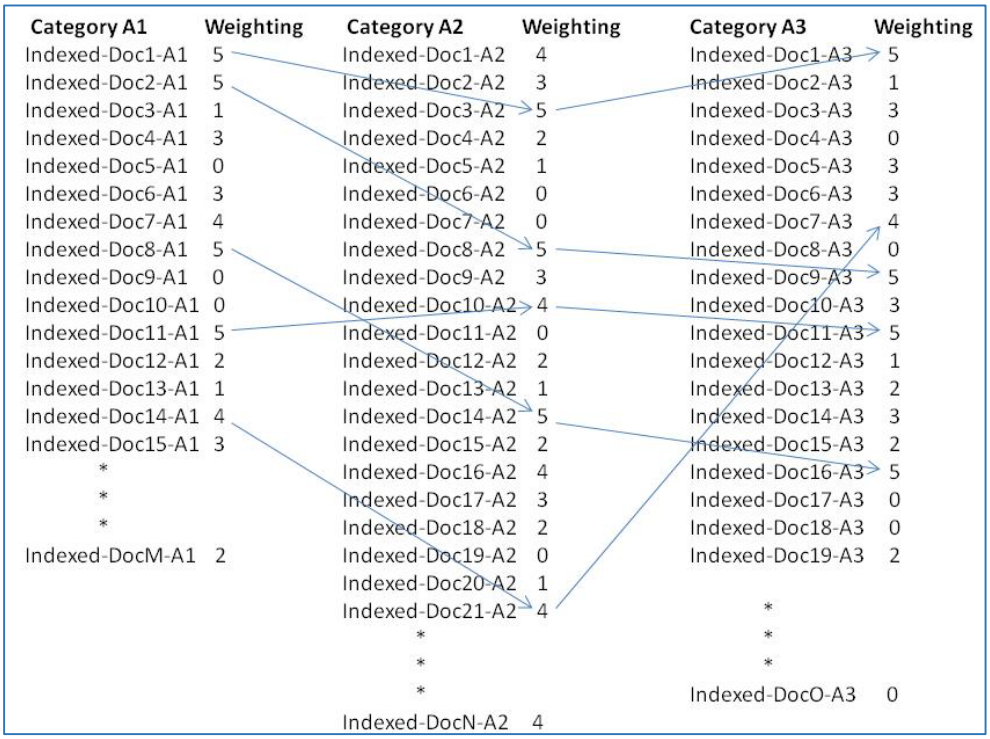

Figure 5.—Mapping proportional weighting scales across all categories.

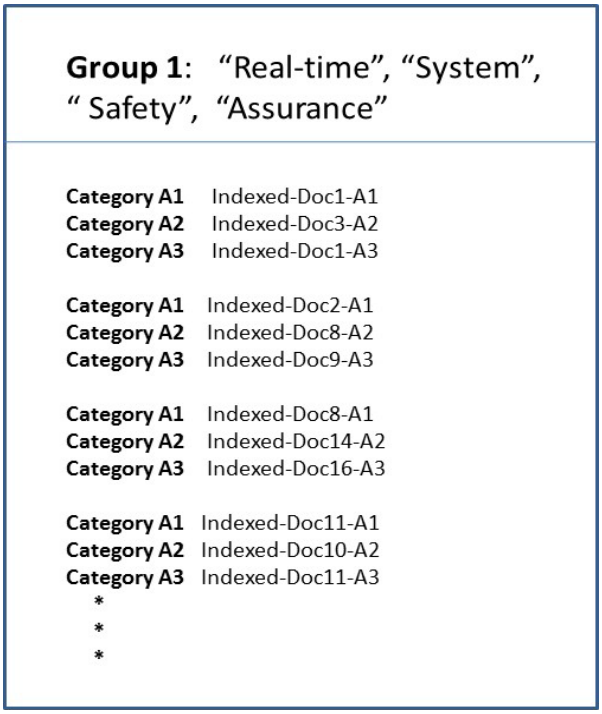

Figure 6.—List of Group 1 categories corresponding with their indexed documents.

The process for matching and ranking is the same for the four combinations of k=3 and k=4. The results of each of these four combinations are stored in Groups 2 to 5 shown in Figure 3.

Experimental Results

The groups are retrieved from the database in the lowest to highest scale order (Groups 1 to 5, resulting in the most relevant documents being listed first. To illustrate the search function, Category A1 is assigned to “SBIR” and Category A2 to “Technology Transfer.” Hundreds of documents are stored in a database that correspond to these two categories. Results of the query terms associated with “diode laser” are shown in Figure 7. In addition to indexing the data associated with documents, the system stores and retrieves the data in its original form so that it populates the search results list with actual data. This feature makes it easier for users to see which documents are most relevant and worth opening for full review. The query terms are also highlighted in the result output displayed on a screen to make the result page more readable.

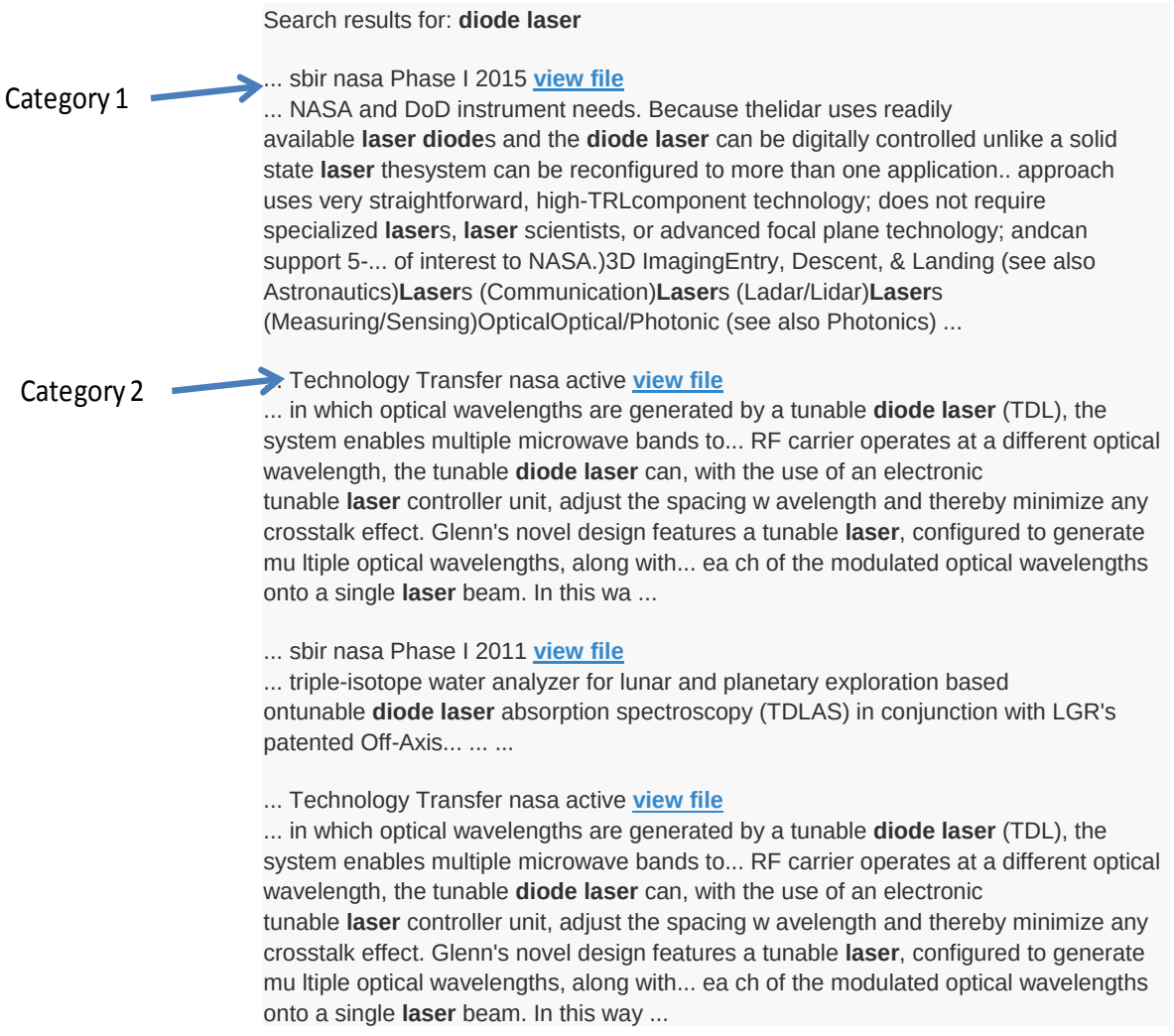

Figure 7.—Search results for the query term “diode laser.

... sbir nasa Phase I 2014 **view file**
SBIR-14-1-S1.07-9837 Unmanned Aerial Vehicle **Diode Laser** Sensor for MethanePOTENTIAL NON-NASA COMMERCIAL APPLICATIONS (Limit 1500 characters,... also Biological Health/Life Support)InfraredForm Generated on 04-23-14 17:37Unmanned Aerial Vehicle **Diode Laser** Sensor for MethaneNASA SBIR 2014 SolicitationFORM B - PROPOSAL SUMMARYPROPOSAL NUMBER:14-1 S1.07-9837SUBTOPIC TITLE:Airborne Measurement SystemsPROPOSAL TITLE:Unmanned Aerial Vehicle **Diode Laser**Sensor for MethaneSMALL BUSINESS CONCERN (Firm Name, Mail Address,... characters, approximately 200 words) A compact, lightweight, and low power **diode laser**

... Technology Transfer nasa active **view file**
Cascading_Tesla_Oscillating_Flow_**Diode**_-20150909-LOW NASA Technology Transfer ProgramBringing NASA Technology Down to EarthTHE... a smaller bias signal. NASA Glenn's cascading Tesla oscillating flow **diode** forces helium gas to flow in predominantly one... the much larger main flow into this tortuous path. While micro-**diode**s have been developed in the past, this innovation casc ades Tesla **diode**s to create a much higher pressure in the gas... createjobs, and improve quality of life. LEW-18862-1240Cascading Tesla Oscillating Flow **Diode**Terrestrial use of Stirling engines is a growing ...

... sbir nasa Phase I 2014 **view file**
... Arraynavigation and landing systems, 3D imaging for docking systems, and **laser** ranging.POTENTIAL NON-NASA COMMERCIAL APPLICATIONS (Limit 1500 characters, approximately 150 words) Military applications include **laser** rangefinding, ladar imaging, autonomous navigation. Commercial applicationsinclude automobile driver... of interest to NASA.)3D ImagingEntry, Descent, & Landing (see also Astronautics)**Laser**s (Guidance & Tracking)**Laser**s (Ladar/Lidar)Materials & Structures (including Optoelectronics)OpticalRanging/TrackingTransmitters/ReceiversForm Generated on 04-23-14 17:37High... need, a near-infrared (NIR) high gain, low excess ...

... Technology Transfer nasa active **view file**
... of products such as electro-static discharge (ESD) compliance testing supplies, **laser** power supplies, radar transmission equipment, high-voltage power supplies, and... ... ...

1 **2** **3** **4** **5** **6** **7** **8** **9** **10** **11** **...** **Next »**

Figure 7.—Concluded.

Conclusion

This report presents an optimal technique to build full-text search engines for searching databases based on users' queries and then selecting documents across each category that have the most relevant documents. This search engine is suitable for Web-based mobile applications and is compatible with all MySQL (MySQL AB) database management systems, making it easy to integrate into a wide variety of commercial database products. A feature that highlights the search terms is also integrated into the search engines to make the results pages more readable.

References

1. Gao, Rujia, et al.: Application of Full Text Search Engine Based on Lucene. Advances in Internet of Things, vol. 2, 2012, pp. 106–109. http://dx.doi.org/10.4236/ait.2012.24013 (Accessed Sept. 27, 2016).
2. Zhong, L.; Wang, H.; Li, R.T., and Song, H.Z.: Research and Development of Full Text Search Engine Based on Lucene. Computer Engineering, http://en.cnki.com.cn/Article_en/CJFDTOTAL-JSJC200604031.htm [Chinese journal].
3. Hellerstein, Joseph M.; Stonebraker, Michael; and Hamilton, James: Architecture of a Database System. Foundations and Trends in Databases, vol. 1, no. 2, 2007, pp. 141–259.
4. Saikia, Amlanjyoti, et al.: Comparative Performance Analysis of MySQL and SQL Server Relational Database Management Systems in Windows Environment. IJARCCE, vol. 4, no. 3, 2015, pp. 160–164. http://www.ijarcce.com/upload/2015/march-15/IJARCCE%2039.pdf (Accessed April 25, 2016).
5. Jun, Sunghae: An Efficient Connection between Statistical Software and Database Management System. IJCSBI, vol. 8, no. 1, 2013. http://www.ijcsbi.org/index.php/ijcsbi/article/viewFile/229/67 (Accessed April 25, 2016).