Semantic Desktop Search Engine using Graph Database

Soumya George, M. Sudheep Elayidom, T. Santhanakrishnan

Abstract: The rise of big data with advancement in technology leads to an ever-increasing demand for a personalized search engine to search the huge amount of data residing in personal computers. A desktop search engine is used to search files or data in a user's personal systems. This paper proposes a graph based semantic desktop search engine, GSDSE that uses the Word Sequence Graph model to store the file details and contents inside a graph database using full text indexing approach. The main features of GSDSE include content-based query autosuggestion based on entire query term sequence, link based page ranking, the semantic search of different query combinations and generation of content based valid search snippet view. To prove the efficiency and reliability of GSDSE, we conduct a comparsion study between Copernic Desktop search engine and GSDSE, and the results proved that the proposed system is efficient concerning efficiency and reliability.

Index Terms: Desktop search engine, Graph database, Word sequence graph model, Semantic search engine.

I. INTRODUCTION

The era of big data increases the amount of data that each user handles. Besides this, advancements of technology lead to new personalized systems with immense storage capacity. All these leads to a high demand for reliable desktop search engine that can search vast amount of data files and folders in a fast and efficient manner. Wide varieties of desktop search engines are available with different features like Copernic, Lookeen, etc. [1]. This paper proposes a graph based semantic desktop search engine, GSDSE based on Word Sequence Graph model, WSG that uses a full text indexing approach to store document or file details and contents inside graph database [2]. Users have the option to filter their search results by file type, folder name, etc. The main features of GSDSE include graph based query autosuggestion based on entire query term sequence - GQAS, page ranking, graph based semantic search of different query combinations - GSSQC and generation of content based valid search snippet view. Graph based document representation enables fast search and retrieval efficiently and reliably by utilizing index free adjacency feature of a graph database.

GSDSE is based on Word Sequence Graph model that uses a graph of word approach where each sentence in each document is stored as a graph of word model by creating a unique node for each document and each unique non-stop word term. Each Document was connected to first non-stop word term of each sentence by an edge of type "contents" and an edge of type "next_seq" was used to connect adjacent non-stop word terms in each sentence. All stop words including symbols and punctuation marks between adjacent non-stop words were concatenated into one string and store in the edge in between as "stop_word" property. Other edge properties include sentence number, sequence id which is the unique document node id, case of succeeding node represented as 'U,' 'S' or 'N' for upper case, sentence case and lower case or no case respectively. All terms in sentence case or upper case were converted to lower case and others will be stored as such. Document node stores properties such as file name, folder name, file type, full path of the file, date, etc. [2].

Tika's AutoDetect Parser was used to parse all types of document files to a plain text format which is then converted to graph based representation of files [3]. Stanford JavaNLP API was used for sentence splitting which in turn converted into words to represent as a sequence of word graph model [4]. This graph based full text indexing approach enables user to search files by contents, and it can be of any length where the main advantage of WSG model lies. Entire file system with all folders and subfolders were indexed. The simple design of index construction of desktop file system is illustrated in Fig.1.

Search engine interface consists of total five panels as shown in Fig. 2. Users can enter their search queries in the text field at panel (1). Users have the options to filter their search results by folder (2) or by file type (3). Matching results will be displayed in the result panel at (4). Search snippet view of matching sentence will also be displayed along with file type icon and file path. Panel (5) is the preview panel used to display search snippet view of the matching sentences of the currently selected file in the result panel with all correct matches highlighted in red color. Clicking a file in the result panel displays its preview details in the preview panel. Double clicking a result or its preview opens the file.

Revised Manuscript Received on December 22, 2018.

Soumya George, Research Scholar, Department of Computer Applications , Cochin University of Science and Technology Kochi-22, India mariyam.george@gmail.com

M. Sudheep Elayidom, Associate Professor, Division of Computer Engineering, Cochin University of Science and Technology Kochi-22, India, sudheepelayidom@hotmail.com

T. Santhanakrishnan. Scientist, Govt. of India, Ministry of Defense, Naval Physical and Oceanographic Laboratory Thrikkakkara, Kochi-21 santhannpol@gmail.com

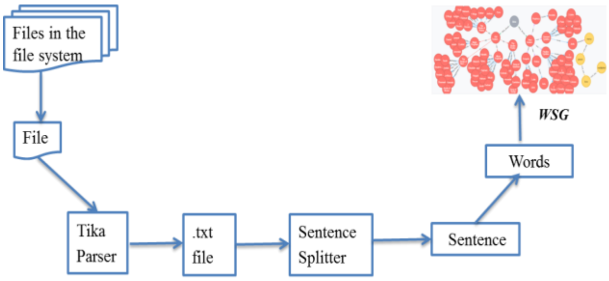

Figure 1: GSDSE Design

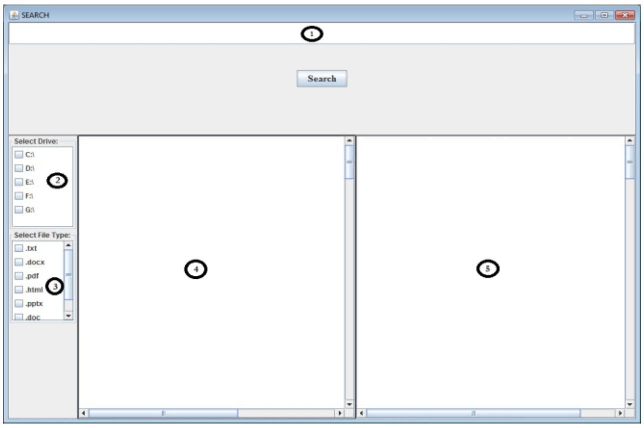

Figure 2: GSDSE Interface

II. GRAPH BASED QUERY AUTO SUGGESTION (GQAS)

One of the striking feature of GSDSE is the Graph based Query Auto Suggestion, GQAS. Sequence word graph representation of documents facilitates faster query auto suggestion to assist users in query completion by providing various suggestions for the next term to enter. Normally query auto suggestion methods use the search history of users to provide suggestions. But GQAS is content based by providing suggestions based on actual contents of documents. Also rather than just relying only on the previous term of query for auto suggestion, GQAS uses the entire sequence of all entered query terms to retrieve the name of all nodes connected to the end node of entered path sequence of query terms having same sentence number and sequence id in the selected path. Fig. 3 represents the screenshots of query auto suggestions for the query “partial order relation and poset.”

III. GRAPH BASED SEMANTIC SEARCH OF QUERY COMBINATIONS (GSSQC)

Graph based data representation enables semantic search of any combinations of the query keywords by applying a combinatorics algorithm by integrating wordnet.

EXPERIMENTAL ANALYSIS AND RESULTS

To evaluate the reliability and efficiency of proposed system, a graph based index was created for a set of different types of documents including .pdf, .ppt, etc. Neo4j graph database was used as storage back end with java programming language. GSDSE performs a semantic search for different query combinations by utilizing Wordnet Java API to find matching results. A search for the different query combinations of the query “Text categorization” is given in Fig.5 by replacing query terms by its synonyms, hyponyms or hypernyms. Copernic desktop search engine is reviewed as the most rated desktop search engine in the market [5][1]. To compare the reliability and efficiency of proposed system, a comparison with Copernic desktop search engine free version is done with the same set of text documents by setting the Copernic index options similar to that of GSDSE. Copernic desktop search engine also search entire document contents instead of searching using file names only as in Locate32 or Everything desktop search engines, which makes it a perfect candidate for comparison with GSDSE. Also, Users can refine their search by different modifiers like file type, folder, etc. in Copernic, but can choose only one folder or one file type at a time. But the word limit of the search engine makes it unable to search for lengthy queries. Copernic is based on traditional bag-of words model which restricts users to use search modifiers or operators to search for multi-word keywords or key phrases. The main disadvantage is that even it searches all files with all keywords using search modifiers, it may not be in correct sequence or word order which distorts the relevance measure used for ranking results. Again, it uses only local history to create query auto suggestions and does not perform semantic search for the documents. Precision and recall values of GSDSE always remain high, and this is represented in Fig. 6 where the no: of results for the same query differ by more than 32 results. Link based Page Ranking algorithm along with semantic search integrity makes GSDSE efficient and reliable to retrieve all true positives. Again, the page ranking algorithm used for Copernic is not reliable as highly relevant results ranked less in retrieval. The results of comparison.

IV. ADVANTAGES AND LIMITATIONS

From the results given in table above, it is shown that GSDSE outperforms when compared to Copernic. The main features or advantages of GSDSE include:Full text search with semantic search enabled. File preview by displaying matching sentences along with exact matches highlighted in red color by preserving the text case of original document. Users have the option to select multiple folders or multiple file types at a time with a user–friendly interface. GSDSE provides query auto-suggestion based on file contents rather than history. No query word limit so that user can search for text of any length Users can filter search by file size, file name or any other parameters stored for each file. The main disadvantages of GSDSE includes: -

Time to index full text in sequence takes more time when compared to Copernic. GSDSE index only text contents. All Images or figures will be discarded. Only case of text is stored. All other formatting options will be discarded. Even though Copernic desktop search engine full version supports almost all features, the starting price for the product is around $50.

V. CONCLUSION

In this paper, we introduced a novel graph based semantic desktop search engine, GSDSE based on Word Sequence Graph (WSG) representation to enable fast and efficient search and retrieval of lengthy queries or long-tail queries including stop words. Important striking features of GSDSE include GQAS - Graph based Query Suggestion, GSSQC - Graph based Semantic Search of Query Combinations, LBPR - Link based Page Ranking, etc. A comparison with Copernic Desktop search engine was done to prove the efficiency of GSDSE when compared to others. The experimental analysis proves that GSDSE outperforms when compared to others in reliability and efficiency. This way of representing documents can be used for many applications like plagiarism detection, Question-Answering systems, etc.

REFERENCES

1. Joel Lee (2016). 10 Best Free Search Tools for Windows 10. http://www.makeuseof.com/tag/10-best-free-search-tools-windows-10
2. Soumya George, M. Sudheep Elayidom and T. Santhanakrishnan (2017). A Novel Sequence Graph Representation for Searching and Retrieving Sequences of Long Text in the Domain of Information Retrieval. IJSRCSEIT Vol. 4 Issue 2
3. Apache Tika API Usage Examples https://tika.apache.org/1.17/examples.html
4. The Stanford NLP Group. https://nlp.stanford.edu/nlp/javadoc/javanlp/edu/stanford/nlp/process/DocumentPreprocessor.html
5. Lu, C. T., Shukla, M., Subramanya, S. H., & Wu, Y. (2007, August). Performance evaluation of desktop search engines. In Information Reuse and Integration, 2007. IRI 2007. IEEE International Conference on (pp. 110-115). IEEE.
6. Rajesh, M., and J. M. Gnanasekar. "Path Observation Based Physical Routing Protocol for Wireless Ad Hoc Networks." Wireless Personal Communications 97.1 (2017): 1267-1289.
7. Rajesh, M., and J. M. Gnanasekar. "Sector Routing Protocol (SRP) in Ad-hoc Networks." Control Network and Complex Systems 5.7 (2015): 1-4.
8. Rajesh, M. "A Review on Excellence Analysis of Relationship Spur Advance in Wireless Ad Hoc Networks." International Journal of Pure and Applied Mathematics 118.9 (2018): 407-412.
9. Rajesh, M., et al. "SENSITIVE DATA SECURITY IN CLOUD COMPUTING AID OF DIFFERENT ENCRYPTION TECHNIQUES." Journal of Advanced Research in Dynamical and Control Systems 18.
10. Rajesh, M. "A signature based information security system for vitality proficient information accumulation in wireless sensor systems." International Journal of Pure and Applied Mathematics 118.9 (2018): 367-387.