Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 89 (2016) 422 – 427

Procedia Computer Science

Twelfth International Multi-Conference on Information Processing-2016 (IMCIP-2016)

A Proposal for Exhaustive Search on Desktop Data

Mamta Kayest* and S. K. Jain

National Institute of Technology, Kurukshetra 136 119, India

Abstract

Data on one's personal computer (PC) is continue to increase day by day and it has made need of managing personal desktop data as an active area of research. Managing desktop data includes an efficient way of searching and retrieving desired data and information from it. Several search engines and search tools are developed to provide search on the desktop data. In this paper, we present a solution for providing exhaustive search on one's heterogeneous desktop data.

Peer-review under responsibility of organizing committee of the Organizing Committee of IMCIP-2016

Keywords: Desktop Data; Desktop Search; Metadata; Partial Content Retrieval; Personal Data Management.

1. Introduction

Managing personal data is an emerging area of research because of the availability of large amount of digital data. Desktop data is a form of personal data available on one's personal computer system. Desktop Search is a specialized instance of the Information Retrieval (IR) technology and provides search over personal data items on one's desktop. Various Desktop Search Engines (DSEs) such as Google[14], Yahoo![23], Corpenic[8], X1[22] manage one's personal desktop data with an extension to email servers. A DSE uses crawler programs to extract data from various data sources that is indexed for further processing to get desired information. In this paper, we design a desktop search system for managing personal desktop data.

The proposed system makes search for files & folders and also retrieves partial contents of a semi-structured and unstructured data files. Here, a semi-structured and unstructured data file refer to an XML file and a text data file respectively. Files are searched on the basis of their properties. A user can search a file on the basis of one or more file's properties such as file name, last access date, last-modified date, creation date, extension and size (KB). For folders one can search on the basis of properties including folder name, creation date, modification date, and access date. Partial content retrieval from semi structured data text files have also been included in the proposal as retrieving partial contents from files is an active research issue in managing desktop data[19].

The next section of the article discusses problem definition and related work briefly, Section 3 and 4 presents a Data Flow Diagram (DFD) and algorithm for the proposed system respectively and Section 5 concludes the paper with future work.

*Corresponding author. Tel.: +91 -8221906216.
E-mail address: mamtakayest.nitkkr@gmail.com

Peer-review under responsibility of organizing committee of the Organizing Committee of IMCIP-2016
doi:10.1016/j.procs.2016.06.090

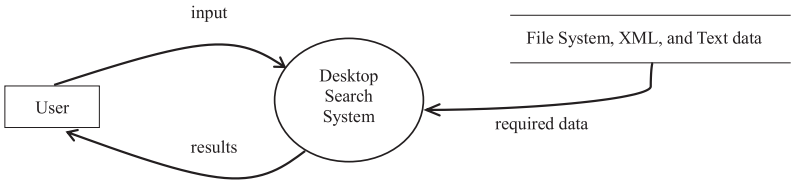

Fig. 1. Context Diagram of the Proposed Desktop Search System.

2. Related Work

Personal data refers to data related to one who accessed, owned and maintained it during his/her lifetime. Personal data consists of a variety of data including images, videos, semi-structured data, text data, emails, and so on. This data may be available in both centralized and distributed forms. Desktop data is an example of centralized personal data; various desktop search engines and tools have been developed for managing and improving search over data. Feldsper[5], Phlat[9], Semex[4], XSearcher[6], etc. are few examples of search system that retrieve information using associations among data. Google desktop search[14], Yahoo![23], Corpernic Desktop Search[8], Windows search[21] and many more desktop search engines are in use for searching data from desktop systems. These search systems have been compared on various parameters[1,3,17]. The DSEs works on the principles of file systems of the underlying operating systems. One of the limitation with the DSEs is that they do not support the retrieval of partial contents from files[19]. To search through a DSE approach, user first enters the search query to the search engine, and the search engine looks up the indexed database for getting desired results[7]. Crawler programs are used by the DSEs for crawling and extracting information. This information is then further used by an indexer to create an indexed database. The problems with DSEs are that they lack in providing partial retrieval of information[19], supports simple keyword queries only; hence, do not support complex queries, no support for semantic integration, and takes initial indexing time once installed. Modelling and querying of heterogeneous desktop data further increases the complexity for developing a desktop search system. It explores a new research dimension for personal data management[2,10,15]. In 2006, a graph data model has been proposed for modeling and providing uniform view over the personal data namely iMeMex Data Model (iDM)[11,12] and to query over the uniform view a query language iMeMex Query Language (iQL) is used, which is complex to understand by a novice; here, it is expected from the users that they have knowledge of the underlying structure of the personal data. Recently in 2015, a framework is introduced for accessing heterogeneous data with different methods for query answering[20]. Similarly, various other methods have been proposed to query over XML data[13,18]. In our previous work[16], we proposed a desktop search system for searching desktop data for various file types and this paper is an extension of our previous work.

3. A Data Flow Diagram and Description of the Proposed System

This section describes the design of the proposed search system in detail. The desktop search system offers metadata based search on files & folders, and content based search on XML and text file. Figure 1 depicts a context diagram of the system which is further decomposed in Fig. 2 and Fig. 3. User submits a query to the desktop search system which in turn interacts with the file system of underlying operating system form required information.

The system processes a query and return results to the user. The first level DFD of the context diagram is shown in Fig. 2. Fig. 3 exhibits second level DFD of the context diagram. Figure 2 contains four modules namely (1) execute file search, (2) execute folder search, (3) execute XML search, (4) execute text search. The names of each module also exhibit the functionalities to be implemented by them.

Figure 3 exhibits the further details of the data and modules as shown by the DFD in Fig. 2. Here, process input data module resolves a query based on its category and forwards to one of the four modules as shown in Fig. 3. Decomposition of input data for the four modules have also shown in Fig. 3. Individual functionality of each module in the DFD shown in Fig. 3 supports a modularization criteria which appears appropriate here. The system offers several search options to the users and the same can be summarized here as follows: