Design of a vertical search engine for synchrotron data: a big data approach using Hadoop ecosystem

Ali Khaleghi¹ · Kamran Mahmoudi¹ · Sonia Mozaffari¹

Received: 25 May 2019 / Accepted: 28 October 2019 / Published online: 28 November 2019
© Springer Nature Switzerland AG 2019

Abstract

A synchrotron as an experimental physics facility can provide the opportunity of a multi-disciplinary research and collaboration between scientists in various fields of study such as physics, chemistry, etc. During the construction and operation of such facility valuable data regarding the design of the facility, instruments and conducted experiments are published and stored. It takes researchers a long time going through different results from generalized search engines to find their needed scientific information so that the design of a domain specific search engine can help researchers to find their desired information with greater precision. It also provides the opportunity to use the crawled data to create a knowledgebase and also to generate different datasets required by the researchers. There have been several other vertical search engines that are designed for scientific data search such as medical information. In this paper we propose the design of such search engine on top of the Apache Hadoop framework. Usage of Hadoop ecosystem provides the necessary features such as scalability, fault tolerance and availability. It also abstracts the complexities of search engine design by using different open source tools as building blocks, among them Apache Nutch for the crawling block and Apache Solr for indexing and query processing. Our primary results obtained by implementing the proposed method in single node mode, the index of over a hundred thousand pages was created with the average fetch interval of 30 days having 28 segments and approximately 570 MB size. The performance factors such as the usage of available bandwidth and system load were logged using Linux's *sysstat* package.

Keywords Synchrotron · Search Engine · Information retrieval · Big data · Hadoop · Solr · Nutch

1 Introduction

1.1 Particle accelerator and synchrotron

Experimental physicists working in different fields of study conduct various experiments in a verity of laboratories. Synchrotron as an experimental physics facility enables scientists to conduct experiments and study materials at a nanoscopic scale. Synchrotron radiation has become an essential part of research in multiple disciplines that depend on light source for their studies [1]. The radiation produced by a synchrotron can be used to study samples

with higher precision through various experiments done in the fields of physics, chemistry, biology, medicine, etc.

Synchrotron experiments can be divided into Spectroscopy, Imaging and Scattering major categories [2].

1.2 Synchrotron data

Synchrotron experiments are conducted at places called Beam-line where the synchrotron radiation is projected to the sample, then using several detectors the experiment data is generated and stored in the data center for further analysis. Depending on the category of the

✉ Ali Khaleghi, akhaleghi@eng.ikiu.ac.ir; Kamran Mahmoudi, kmahmoudi@edu.ikiu.ac.ir; Sonia Mozaffari, soniamozaffari@edu.ikiu.ac.ir |
¹Imam Khomeini International University, Qazvin, Iran.

experiment a notable amount of data is generated specially for the Imaging category. The data generated by detectors at CERN particle accelerator is expected to be around 50–70 TB at 2018 [3]. This huge amount of data is just for the experiments at LHC as a part of synchrotron community. The usage and development of Big Data tools is extremely required for analysis of such data. One of the important needs of scientists working at synchrotrons is the ability of searching various datasets and documents related to their specific topic of research. In this paper we will propose a design for a domain specific search engine as a solution for the need above. Synchrotrons share a lot of documents and data which are related to their design and experiments. These documents are useful for scientists who are designing this facility such as the researchers at Iranian Light Source Facility (ILSF). The published data on Synchrotron websites are also useful for beamline scientists and also scientific directors of other Synchrotrons around the world. One of the main problems here is the difficulty in finding desired information. In this paper we propose the design of a domain specific search engine to address this need. The rest of the paper is organized as follows: Sect. 2 presents related work, Sect. 3 presents methodology, primary results are presented at Sect. 4 and finally, Sect. 5 holds our conclusions.

2 Background

A domain specific search engine creates a searchable index of contents related to a particular subject. Due to huge amount of data published on the internet, design of such search engines can help us to find more accurate data easier than using a general search engine. The Sect. 1.1 introduce the three main modules of a search engine then in Sect. 1.2 we will name several use cases that a domain specific search engine has been designed. Finally, in the Sect. 2.1 the architecture of a search engine is presented and a vertical search engine named HVSE which is built

using the Hadoop framework with a similar approach to the current paper is introduced.

2.1 Domain specific search engines

Widyantoro and Yen [4] introduce a domain specific search engine for searching academic papers' abstracts using a fuzzy ontology method for query refinement. For searching academic papers in the field of particle accelerator, CERN has used FAST search engine which is introduced in 2007 and is a subsidiary of the Microsoft since 2008, this search engine enables researchers to eliminate unwanted results using full Boolean queries [5]. As another example for domain specific search Mišutka and Galamboš [6] proposed a method for searching the mathematical content that can be adopted by any full text search engine.

Researchers use Google Scholar to find academic papers, but an important issue with Google Scholar is the lack of custom search technique, in [7] a domain specific search engine is introduced that uses a new search methodology called n-paged-m-items partial crawling algorithm that is a real-time faster algorithm, authors of the paper reported a better performance comparing with Google Scholar.

Because most of medical queries are long and finding relevant results are also difficult, Luo et al. proposed a special search engine for medical information to simplify medical searching, which called Medsearch. It splits long queries to several short queries and then finds more relevant results for that [8]. Search engine design.

A vertical search engine called HVSE has been proposed in [9], in which the authors have improved topic oriented web crawler algorithms and developed a search engine based on Hadoop platform. With the decentralized Hadoop platform this search engine can have higher efficiency for massive amount of data due to the ability of expanding the Hadoop cluster.

The architecture of a search engine consists of four main parts as shown in Fig. 1. The crawler as the first part which is responsible for collection of data from web pages. The second part is the indexer which creates a

Fig. 1 Architecture of a search engine

searchable index of the collected raw data. The third part is the query parser which parse the user's input query and retrieves the related information. The last and forth part is the user interface which could be in the form of a web application or mobile app that facilitates the search and shows the results to the end user.

3 Methodology

As mentioned in previous sections, one of the problems of scientists working in synchrotron facilities is finding related documents and datasets for their research. It is difficult to use a general search engine to find specific scientific data and to our knowledge there has not been any domain specific search engine created for the field of particle accelerator physics. The synchrotron information come from various sources, most of which is publicly accessible through the websites of different light sources and laboratories. Some other sources of valuable information is the facilities data center that stores experimental data, also each facility have their information system for storing the status of devices and infrastructure.

In this section we propose an architecture of a domain specific search engine that can be used as a solution to the above issue. The architecture includes an implementation of Apache Hadoop framework and HDFS as the basis for setting up the search engine. Apache Nutch and Solr will be deployed over the Hadoop framework which will be used for crawling and indexing respectively as shown in Fig. 2.

3.1 Hadoop ecosystem

Hadoop is an apache projects which is developed using Java as a framework for processing Big Data. With the implementation of a Distributed File System and Map-Reduce as a

programming model [10]. In our architecture Hadoop HDFS is used by the Apache Solr to store the index the documents that are retrieved by Apache Nutch which is an open-source web crawler software that is used for crawling websites. With Nutch, you can create your own search engine easily and customize it based on your needs [11].

3.2 Crawler module

Apache Nutch which is an extensible and scalable web crawler is used for crawling synchrotron websites to collect the required data. It can be configured to use HDFS providing the scalability for storing huge amount of data using the distributed file system. Nutch has a number of plug-ins that can be used to process a variety of file types such as Plain Text, XML, Open Document, Microsoft Office, PDF, RTF etc. This feature is very useful since it can be used for processing most of the Synchrotron Data formats.

Most researchers design their own implemented web crawlers when it comes to creating a focused crawler. Supervised and semi-supervised machine learning techniques are used to improve relevant document retrieval but there is little literature on the performance of such crawlers for web-scale data. In order to crawl and store specific domains we need to define some URL filters using regular expressions. Figure 3 shows such configuration added to the end of regex-urlfilter text file stored in the Nutch's configuration directory.

3.3 Indexing and query parsing modules

For indexing the fetched documents, Nutch can be configured to send them to Apache Solr which is built on the Apache Lucene information retrieval library. Apache Lucene uses a vector space model method for creating the index and it supports a variety of query types such as fielded term

Fig. 2 Architecture of proposed search engine

Fig. 3 Regular expression used to define URL filters

```

1 +^https?:/([a-z0-9-]+\.)*albasynchrotron\.es/
2 +^https?:/([a-z0-9-]+\.)*elettra\.eu/
3 +^https?:/([a-z0-9-]+\.)*esrf\.eu/
4 +^https?:/([a-z0-9-]+\.)*als\.lbl\.gov/
5 +^https?:/([a-z0-9-]+\.)*aps\.anl\.gov/
6 +^https?:/([a-z0-9-]+\.)*ansto\.gov\.au/
7 +^https?:/([a-z0-9-]+\.)*helmholtz-berlin\.de/
8 +^https?:/([a-z0-9-]+\.)*lightsource\.ca/
9 +^https?:/([a-z0-9-]+\.)*chess\.cornell\.edu/
10 +^https?:/([a-z0-9-]+\.)*diamond\.ac\.uk/
11 +^https?:/([a-z0-9-]+\.)*lnls\.cnpem\.br/
12 +^https?:/([a-z0-9-]+\.)*maxiv\.lu\.se/
13 +^https?:/([a-z0-9-]+\.)*photon-science\.desy\.de/
14 +^https?:/([a-z0-9-]+\.)*nsrrc\.org\.tw/
15 +^https?:/([a-z0-9-]+\.)*pal\.postech\.ac\.kr/
16 +^https?:/([a-z0-9-]+\.)*bnl\.gov/
17 +^https?:/([a-z0-9-]+\.)*www-ssl\.slac\.stanford\.edu/
18 +^https?:/([a-z0-9-]+\.)*sesame\.org\.jo/
19 +^https?:/([a-z0-9-]+\.)*kek\.jp/
20 +^https?:/([a-z0-9-]+\.)*slri\.or\.th/
21 +^https?:/([a-z0-9-]+\.)*psi\.ch/
22 +^https?:/([a-z0-9-]+\.)*synchrotron\.uj\.edu\.pl/

```

with boosts, wildcards, fuzzy (using Levenshtein Distance), proximity searches and Boolean operators.

3.4 User interface

When data is collected by the Apache Nutch and an Index is generated, users can describe their intent in the form of a Query including several keyword terms. The user's input query then should be sent to the search engine to retrieve related information. This can be done through creating different user interfaces to capture the query input. Apache Solr can be accessed through sending a HTTP request containing the user query in the form of field and value. When a query is sent Solr would run the query and send the

results in JSON format. This feature enables the UI designer to create a variety of applications for end users of different platforms such as web pages or mobile applications.

4 Primary results

The proposed architecture was implemented in a single node mode, it indexed about a hundred thousand documents with average fetch interval 30 days creating an index with 28 segments and 579 MB size. The performance of the machine was monitored and logged via sysstats package and KSar. Figure 4 shows the run queue and average load status of the system in 1 h period

Fig. 4 Load of the system during 1 h period of the crawl process

Fig. 5 Network performance during 1 h period of crawling process

Fig. 6 CPU frequency status for all cores

during the crawl process. The network traffic of the eth0 interface is shown in Fig. 5, the Fig. 6 illustrates the CPU status and the sockets status is shown in Fig. 7.

The performance of a crawling system depends on several variables including the load of the system, available bandwidth and network, type and size of crawled items, number of threads etc., as shown in Fig. 4, the average

system load is normally lower than 0.5 while the data is being fetched and the load is increased during the index creation process done by apache Solr. The number of threads and file size limits used to crawl are defined using the default values by Apache Nutch. While we are crawling a limited number of websites politeness along with uneven distribution of URLs can be a limiting factor since other threads should be idle while a thread is downloading an item or processing it relating to a specific host. Here we reach an average of more than 200 Ki as shown in Fig. 5. The number of sockets created during the crawling process and related TCP wait time is shown in Fig. 7, the wait time mechanism keeps the socket open after being shut down by a process for a limited time to prevent the packets being accepted by another process and to ensure the remote end has closed the connection, it is normal for sockets to accumulate when the server is opening and closing sockets with high rates.

5 Conclusion

Due to the vast amount of information published on the web, general search engines have less efficiency while searching for very specific data such as scientific information. Data integration methods such as web data integration can be used to provide means for running complex queries on various integrated data sources. Here we propose a search engine just for indexing webpages of different Synchrotrons for keyword searching. Creation of vertical crawlers is a solution to provide easy and precise search tool. There have been various vertical search engines used for searching scientific papers or medical records, etc., but according to our knowledge a vertical search engine in the field of accelerator physics and Synchrotron community has not been developed. In this paper we proposed a domain specific search engine for being used by scientists working in the field of particle physics and related disciplines in experimental physics facilities to find desired information and datasets with more precision. We reviewed the literature on designing such search engines for various applications then we presented our proposed architecture using the open source tools developed by the Apache software foundation. We used Apache Nutch for crawling different synchrotron data sources such as synchrotron websites and their local repositories. By deploying an Apache Solr instance we will be able to run queries and index the collected data using Apache Lucene. These tools are configured to run over a Hadoop cluster that provides the scalability and fault tolerance required in the design of a search engine. As a future work we will implement our proposed design using a Hadoop cluster with three nodes having an overall 200 CPU cores, 280 GB

Fig. 7 Sockets status during 1 h period of the crawl process

memory and 9 TB of disk space to be used by scientists working at Iranian Light Source Facility (ILSF) and other laboratories of the Synchrotron community worldwide.

Compliance with ethical standards

Conflict of interest On behalf of all authors, the corresponding author states that there is no conflict of interest.

References

- Rahighi J et al (2013) ILSF, a third generation light source laboratory in Iran. TUOAB202, IPAC 13
- Alizada S, Khaleghi A (2017) The study of big data tools usages in synchrotrons. In: Proceedings of 16th international conference on accelerator and large experimental control systems, ICALEPCS2017, Barcelona, Spain (2017)
- CERN About. <http://wlcg-public.web.cern.ch/about>. Accessed 5 Dec 2018
- Widyantoro DH, Yen J (2001) A fuzzy ontology-based abstract search engine and its user studies. In: The 10th IEEE international conference on fuzzy systems, 2001, vol 3. IEEE
- Particle accelerator conference proceedings. <https://accelconf.web.cern.ch/accelconf/JACoW/proceedingsnew.htm>. Accessed 5 Dec 2018
- Mišutka J, Galamboš L (2008) Extending full text search engine for mathematical content. Towards Digital Mathematics Library, Birmingham, United Kingdom, 27 July 2008, pp 55–67
- Saha TK, Shawkat Ali ABM (2013) Domain specific custom search for quicker information retrieval. Int J Inf Retr Res 3(3):26–39
- Luo G, Tang C, Yang H, Wei X (2008) MedSearch: a specialized search engine for medical information retrieval. In: Proceedings of the 17th ACM conference on information and knowledge management, CIKM 2008, Napa Valley, California, USA, October 26–30, 2008
- Lin C, Yajie M (2016) Design and implementation of vertical search engine based on Hadoop. In: 2016 eighth international

- conference on measuring technology and mechatronics automation (ICMTMA). IEEE
10. Zalte SA, Takate VR, Chaudhari SR (2017) Study of distributed file system for big data. *Int J Innov Res Comput Commun Eng* 5(2):1435–1438
 11. Laliwala Z, Shaikh A (2013) *Web crawling and data mining with Apache Nutch*. Packt Publishing, Birmingham

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.