Journal of Multidisciplinary Engineering Science and Technology (JMEST)
ISSN: 2458-9403
Vol. 6 Issue 2, February - 2019

A Data Mining Infrastructure for Cheminformatics

Jungkee Kim
Department of Glocal IT
Sungkonghoe University
Seoul, Korea
jake@skhu.ac.kr

***Abstract*—An enormous increase of data sources for chemical information and biological science requires a new development methodology for mining useful information. Such data sources give us an opportunity to utilize computational tools to mine useful information and to find new patterns in data sets that explain scientific phenomena not yet known. It is also important that non-expert users can access the latest cheminformatics methodology and models to spread the new discoveries. We present our previous developments in cheminformatics procedures and infrastructure that provide an appropriate approach to mining large chemical datasets. We also discuss the limitation of previous challenge and propose a new infrastructure with the state-of-the-art techniques expected to improve the performance.**

Keywords— Cheminformatics; Work flow; Web service; Big Data

I. INTRODUCTION

Recent progress in chemistry and life sciences have led to a large volume of new data sources called Big Data. Big Data is a data set that cannot be proficiently handled by conventional data processing technics. When we classify big data, different principles should be considered in data mining [1]. Cheminformatics belongs to a multidisciplinary filed that integrates life science, chemistry and computer science. The in silico illustration of chemical structures employs particular formats such as XML-based Chemical Markup Lange (CML), SMILES, SDF, and so on. The data in those formats are frequently used in large chemical databases such as PubChem [2], ChEMBL [3], and BindingDB [4]. Therefore, we can access a large volume of chemical compounds and biological activities in a diversity of biological assays.

We need to connect chemical structures to the life science information. For example, systems biologists study the complex biological systems that integrate microarray datasets to biological pathways, using a large number of other data sets to provide evidence for the links [5].

A typical method to access data is a traditional query to the database manage systems by a human. A software agent can access and process the data in a uniform manner without human intervention. Web services are techniques of aggregating and integrating data sources and software. They allocate software applications and data source to be published on the network, therefore making tools and data broadly available with a standardized interface and enabling the construction of application that use distributed resources and data to resolve complex tasks. There are three standards to create Web services. Web Service Description Language (WSDL) is an XML-based standard for presenting Web services and their parameters. Simple Object Access Protocol (SOAP) provides the envelope existing applications to match abstract interfaces in WSDL to their actual executions. Universal Description, Discovery and Integration (UDDI) provokes the publishing and browsing of Web services by user groups. Representational State Transfer (REST) architectural style replaces WSDL since the REST-based design methodology [6] was emerged. In RESTful style, there is no standard such as SOAP and any other payload formatted in HTML, XML, JavaScript Object Notation (JSON), or another format. The aspect of connections between distributed resources is important because it is easy to collect information from a diversity of high throughput screening and vendor catalogues.

The MapReduce framework [7] provides a programming model for parallel and distributed handling of batch jobs on a large number of computing nodes. Each job in the MapReduce divided into two phases –map and reduce. The map phase divides the input data by relating each element with a key. The reduce phase handles each split independently, and all data is processed based on key-value pairs. The map function processes a certain key-value pair and produces a certain number of new key-value pairs. The reduce processes all intermediate values grouped by the same key into another set of key value pairs as output.

A scientific workflow is a specialized form of the general workflow, which designed particularly to compose and implement a set of tasks in an order depending on their relations in a scientific application [8]. The technic of scientific workflow has been successfully applied to the scientific fields including cheminformatics and life science. Scientific Workflow Management System (SWfMS) is a tool to implement workflows and handle data sets. Several Grids workflow projects are developed. Triana [9], Kepler [10], and Taverna [11] are typical examples. Triana is started from a single platform but supports distributed services with Grid awareness. Kepler is also started from a single platform and it fully supports Grid

environment. It is widely used in many scientific domains and provides graphical user interface to organize workflows intuitively. Taverna is part of myGrid project and focuses on applications of life science. It recognizes the importance of provenance and semantics by a textual language.

A workflow scheduler is critical for the efficient workflow management system. Many scientific workflow management systems hire their own scheduling algorithms [12, 13]. We need to find a proper algorithm for a good performance.

The rest of this paper is organized as follows. Section 2 describes our previous work. Section 3 presents MapReduce framework and scheduler. We illustrate a new architecture for data mining of large data sets in Section 4. We summarize and conclude in the last Section.

II. Previous Work

We developed an infrastructure of Web service for cheminformatics that simplifies the access to drug discovery information and the computational techniques that can be applied to it [14]. At that time, the Web services were mostly based on Java. Using Java allows us to deploy our Web services in a Tomcat application module, which allows us to easily support a variety of services and provide an integration with our execution environments. The services themselves are hosted on a diversity of servers and are generally separated from database and functionality. Therefore Web services that provide database functionality will connect to a remote database server to retrieve results.

We implemented Web service wrappers for several free and commercial cheminformatics tools. The commercial tools that we were permitted to use tools such as Digital Chemistry Divisive K-Means for clustering. We had a working relationship with the Murray-Rust group at Cambridge University [15] that was one of sites that had semantic Web approaches to cheminformatics. We implemented several of their Web services such as InChIGoogle, InChIServer, CMLRSSServer, and OSCAR for automatic mining of chemical structure information from documents. We also implemented a large amount of the functionality of the Chemistry Development Kit (CDK) as Web services such as descriptor calculation, 2D similarity and fingerprint calculations, and 2D structure depiction. We experimented a special modified Web service implementation of ToxTree [16] for toxicity flagging.

Web services can be used in workflow tools such as Taverna workbench. The tools allow the creation of new functionality by linking together services and other application and data resource into workflows. Figure 1 illustrates an example of Taverna workbench in a graphical user interface. The interface encloses a list of processes that the user enables invoking on that service. After selecting an operation, the user is accessible with an interface for the operation, which enables the user to specify all the input parameters to the operation. And the user can invoke the operation on the service and obtain the output results.

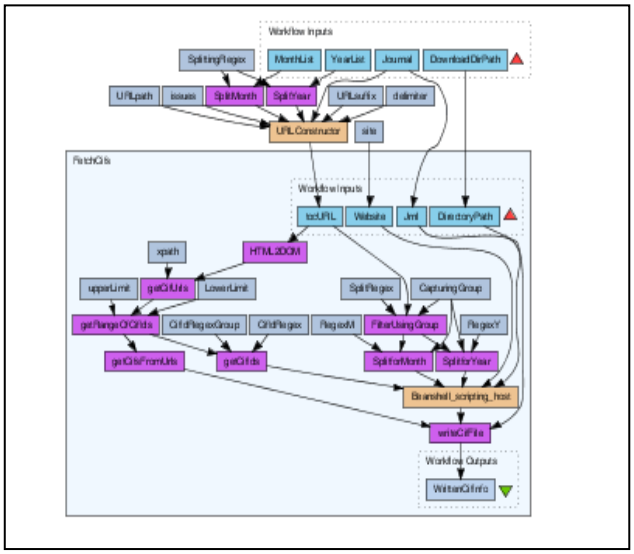

Fig. 1. *CDK Workflow in Taverna workbench.*

III. MapReduce Framework and Scheduler

MapReduce frameworks execute much better in tough environments than other tightly coupled distributed programming frameworks such as Message Passing Interface (MPI) because of their fault tolerance capabilities [17]. They are suitable to support many scientific use cases, and many scientists can build large data-oriented applications easily under could computing environment.

Apache Hadoop [18] is a framework that provides distributed processing of large data sets and the implementation is based on Google MapReduce [19]. The Hadoop Distributed File System (HDFS) follows write-one-read-many pattern and does not support functions to change an existing file. The HDFS is designed for deployment on unreliable clusters and succeeds in reliability by the replication of data files. The Hadoop minimizes data communication by processing computations near the place it is stored. The architecture consists of a master node with many worker nodes and uses a queue for task scheduling and succeeds in load balance naturally among computing tasks.

A classical workflow for collecting related data and inserted into a local database management system (DBMS) before processing data. The HDFS replaces the local database for a temporary storage. Apache Hadoop framework is a promising system to store the extracted huge datasets from databases.

A scheduler of scientific workflows allocate tasks mapping on heterogeneous and distributed resources. A good algorithm can make tasks allocated to the proper resources and arrange the best sequence of parallel tasks. We need to consider two groups – users and service providers. Users are concerned with reliability and the service quality. So they want the result within the proper time. However, the servers aim

at their efficiency to capture maximum revenues. We can consider several strategies such as execution time-based strategy, just-in-time strategy, linear scheduling, policy base strategy, virtual machine strategy, gossip based strategy, reservation based strategy, and heuristic based strategy. We need to optimize our scheduler among those strategies in the future work.

IV. Architecture for Data Mining of Large Chemical Datasets

In our previous work [14, 20], we introduced a chemical mining process to collect chemical structures. Figure 2 presents the architecture of the process implemented on a supercomputer with Message Passing Interface (MPI). Using PHP script queries and PubMed ID, we collect abstracts of research papers in the first step. A group of the abstract text files are assigned to each node in a supercomputer. In a node, a series of batch processes extracts chemical compounds and their three dimensional structures.

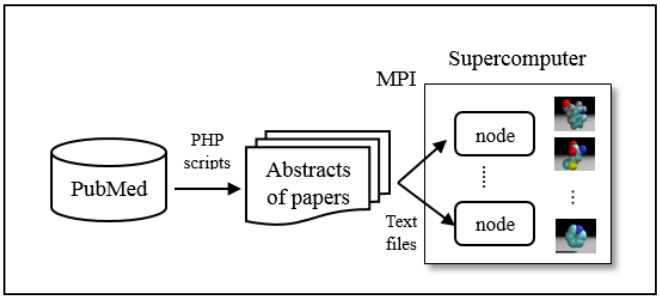

Fig. 2. *Architecture of a chemical compound mining process.*

In the experiment, even the super computer system took a lot of time to process about 500,000 abstracts. We suggest an architecture in which the Hadoop MapReduce Framework replaces the super computer system with a simple MPI. Figure 3 illustrates a new architecture replacing the super computer system in the Figure 2. The input text files are stored in the database (HDFS). The server provides graphical user interface as a part of workflow bench such as Tarverna. Workers are instances of the server and are only accessed by a scheduler that assigns execution tasks for mapping or reducing. We can employ a SOAP library to allow consumption of Web services.

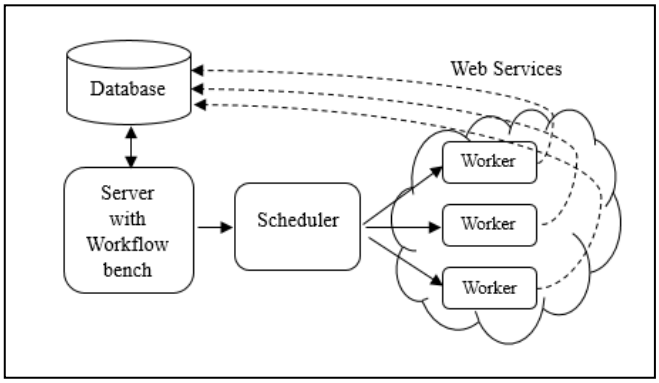

Fig. 3. *New architecture for mining process*

V. Conculusion

With recent progress in chemistry and life science generating a large datasets forces many requirements on a storage and an analysis framework. We describe a review of distributed systems designed to process chemical information. We also present our Web service and workflow development for cheminformatics in the previous work. However, the case study for mining chemical compound demonstrates a need for more efficient architecture for processing large datasets in chemistry and life science field. Thus we propose a new architecture with MapReduce framework to expect to address the performance problem.

References

[1] Y. Hu and J. Bajorath. "Entering the 'big data' era in medicinal chemistry: molecular promiscuity analysis revisited," Future science OA, vol.3, 2017.

[2] E. Bolton, et al., "PubChem: integrated platform of small molecules and biological activities," Annual reports in computational chemistry, Vol. 4. Elsevier, pp.217-241, 2008.

[3] T. Liu, et al.. "BindingDB: a web-accessible database of experimentally determined protein–ligand binding affinities," Nucleic acids research, vol.35, pp. 198-201, 2006.

[4] A. Gaulton, et al., "ChEMBL: a large-scale bioactivity database for drug discovery," Nucleic acids research, vol. 40, D1100-D1107, 2011.

[5] Y. Saeys, I. Inza, and P. Larranaga. "A review of feature selection techniques in bioinformatics," Bioinformatics, vol. 23, pp.2507-2517, 2007.

[6] R. Fielding and R. Taylor, "Principled design of the modern Web architecture," ACM Transactions on Internet Technology (TOIT), vol. 2, pp. 115-150, 2002.

[7] J. Dean and S. Ghemawat. "MapReduce: simplified data processing on large clusters," Communications of the ACM, vol. 51.1, pp. 107-113, 2008.

[8] J. Liu et al. "A survey of data-intensive scientific workflow management," Journal of Grid Computing, vol. 13.4, pp.457-493, 2015.

[9] I. Taylor, M. Shields, I. Wang, and A. Harrison, “The Triana Workflow Environment: Architecture and Applications,” Workflows for e-Science, Springer, pp. 320-339, 2007.

[10] D. Pennington, D. Higgins, A. Peterson, M. Jones, B. Ludascher, S. Bowers, “Ecological Niche Modeling Using the Kepler Workflow System,” Workflows for e-Science, Springer, pp. 91-108, 2007.

[11] T. Oinn, P. Li, D. Kell, C. Goble, A. Goderis, M. Greenwood, D. Hull, R. Stevens, D. Turi, and J. Zhao, “Taverna / myGrid: aligning a workflow system with the life sciences community,” Workflows for e-Science, Springer, pp. 300-319, 2007.

[12] G. Gharooni-farda, F. Moein-darbari, H. Deldari, and A. Morvaridi, Scheduling of scientific workflows using a chaos-genetic algorithm. Procedia Computer Science, vol. 1, pp144501454, 2010.

[13] M. A. Rodriguez and R. Buyya, "Deadline based resource provisioningand scheduling algorithm for scientific workflows on clouds," IEEE transactions on Cloud Computing, vol. 2, pp. 222-235, 2014.

[14] X. Dong, et al. "Web service infrastructure for chemoinformatics." Journal of chemical information and modeling, vol. 47, pp. 1303-1307, 2007.

[15] Murray-Rust Research Group, World Wide Web, http://www-pmr.ch.cam.ac.uk.

[16] ToxTree, World Wide Web, http://sourceforge.net/projects/toxtree.

[17] T. Gunarathne, et al. "Cloud computing paradigms for pleasingly parallel biomedical applications," Proc. of ACM Int. Symp. on HPDC, ACM, pp. 460-469, 2010.

[18] Apache Hadoop, World Wide Web, http://hadoop.apache.org/.

[19] J. Dean, and S. Ghemawat, "MapReduce: simplified data processing on large clusters," *Communications of the ACM*, vol. 51, pp. 107-113, 2008.

[20] J. Kim, "Chemical Compound Mining for Big Data," Proc. of Intl. Conf. on Future Generation Information Tech., 2019.