International Journal On Engineering Technology and Sciences – IJETS™
ISSN(P): 2349-3968, ISSN (O): 2349-3976
Volume III, Issue X, October- 2016

Cloud Service Discovery and MapReduce Framework for Healthcare Data Classification

Ms. R. Saranya, MPhil., Research Scholar,
Dr. N. Chandrakala, MSc., MPhil., PhD., Head/Department of Computer Science.,
SSM College of Arts & Science, Komarapalayam, Tamilnadu, India

Abstract

The cloud services are provided by the service providers to execute user workloads. Cloud resource sharing framework is divided into two types with reference to its access parameter levels. The public cloud environment provides the cloud services to everyone. The private cloud services are employed to share the big data values with limited users only. The hybrid cloud or cross clouds are builds with the combination of the public cloud and private cloud services. The private cloud data values are passed to the public cloud services for the analysis purpose.

Big data analysis operations are carried out under the hybrid cloud environment. The service discovery methods are applied to identify the suitable service providers with reference to the performance parameters. The access history details are utilized for the service selection process. Privacy is required for the private cloud data access history values. The service selection is carried out using History record based Service optimization method (HireSome-II). The history records are partitioned using the K-means clustering algorithm. The privacy preservation approaches are adapted to protect the history record values. The service discovery operations are carried out on the privacy preserved history data values.

The healthcare data classification is carried out on the big data values with history record based service discovery process. Thyroid diagnosis data and history values are protected with anonymization techniques. The MapReduce mechanism is employed to handle the thyroid data classification with privacy features. Thyroid diagnosis data classification is achieved with service discovery and privacy ensured MapReduce mechanism.

1. Introduction

Cloud computing is an emerging paradigm which enables customers to use resources as services. Three prominent delivery models supported by cloud computing are: (i) Infrastructure-as-a-Service (IaaS), (ii) Platform-as-a-Service (PaaS), and (iii) Software-as-a-Service (SaaS). One of the popular offerings of the SaaS cloud is online collaboration service. Online collaboration tools enable various organizations to share and access information in a faster and more seamless manner. A few examples of such service include document centric collaboration, project management, blogs, micro blogging, wikipages, feeds from social networks, file sharing and synchronization, and so on. A common concern in a collaborative environment is that the integrity of data, shared across multiple users, may be compromised. Moreover, choosing an ideal vendor to provide secure and guaranteed collaboration service is also non-trivial. Nevertheless, collaboration has become not only valuable but also essential as it allows the organizations to easily connect with partners, customers, and employees from remote locations with less communication latency. A recent Forrester1 survey found that more than 56% of software decision-makers are using or expected to use Software-asa- Service (SaaS) offerings to replace or complement their existing collaboration technology. Cloud-based collaboration provides storage spaces to multiple organizations where they can share resources, and access them through policies formed by customizing the available APIs. Some noteworthy collaboration service providers2 are: (i) Acrobat.com, (ii) Box.net, (iii) CubeTree, (iv) HyperOffice, (v) Google Apps, (vi) MS Office Live, (vii) Zoho and so on. Collaboration among multiple domains is either tightly or loosely-coupled:

Tightly-coupled collaboration: In this collaboration, there exists a master domain which mediates accesses to individual local domains through a global policy which is formed by integrating policies from participating domains. In such collaboration, interoperation needs are static and predefined. Loosely-coupled collaboration: In this collaboration, independent systems dynamically come together to share information for a period of time. No global policy is maintained as interoperation requests are "on-demand" to facilitate dynamic data sharing. In a cloud environment, both tight and loose coupling may take place depending on the nature of collaboration. For example, if different departments of an organization collaborate using cloud services, it is an instance of tightly coupled collaboration. If autonomous domains collaborate

"ondemand" for a limited period of time, it is an example of loosely-coupled collaboration.

2. Related Work

The handiest option for handling data distributed across several datacenters is to rely on the existing cloud storage services. This approach allows transferring data between arbitrary endpoints via the cloud storage and it is adopted by several systems in order to manage data movements over wide-area networks. Typically, they are not concerned by achieving high throughput, nor by potential optimizations, let alone offer the ability to support different data services. Our work aims is to specifically address these issues.

Besides storage, there are few cloud-provided services that focus on data handling. Some of them use the geographical distribution of data to reduce latencies of data transfers. Amazon's CloudFront, for instance, uses a network of edge locations around the world to cache copy static content close to users. The goal here is different from ours: this approach is meaningful when delivering large popular objects to many end users. It lowers the latency and allows high, sustained transfer rates. Similarly, considered the problem of scheduling data-intensive workflows in clouds assuming that files are replicated in multiple execution sites. These approaches can reduce the makespan of the workflows but come at the cost and overhead of replication. In contrast, we extend this approach to exploit also the data access patterns and leverage a cost/performance tradeoff to allow per file optimizations of transfers. The alternative to the cloud offerings are the transfer systems that users can choose and deploy on their own, which we generically call user-managed solutions.

A number of such systems emerged in the context of the GridFTP transfer tool, initially developed for grids. In these private infrastructures, information about the network bandwidth between nodes as well as the topology and the routing strategies are publicly available. Using this knowledge, transfer strategies can be designed for maximizing certain heuristics; or the entire network of nodes across all sites can be viewed as a flow graph and the transfer scheduling can be solved using flow-based graph algorithms. However, in the case of public clouds, information about the network topology is not available to the users. One option is to profile the performance. Even with this approach, in order to apply a flow algorithm the links between all nodes need to be continuously monitored. Such monitoring would incur a huge overhead and impact on the transfer. Among these, the work most comparable to ours is Globus Online, which provides high performance file transfers through intuitive web 2.0 interfaces, with support for automatic fault recovery.

Globus Online only performs file transfers between GridFTP instances, remains unaware of the environment and therefore its transfer optimizations are mostly done statically. Several extensions brought to GridFTP allow users to enhance transfer performance by tuning some key parameters: threading or overlays. Still, these works only focus on optimizing some specific constraints and ignore others. This leaves the burden of applying the most appropriate settings effectively to users. In contrast, we propose a self-adaptive approach through a simple and transparent interface, that doesn't require additional user management.

Other approaches aim at improving the throughput by exploiting the network and the end-system parallelism or a hybrid approach between them. Building on the network parallelism, the transfer performance can be enhanced by routing data via intermediate nodes chosen to increase aggregate bandwidth. Multihop path splitting solutions replace a direct TCP connection between the source and destination by a multi-hop chain through some intermediate nodes. Multi-pathing employs multiple independent routes to simultaneously transfer disjoint chunks of a file to its destination. These solutions come at some costs: under heavy load, per-packet latency may increase due to timeouts while more memory is needed for the receive buffers. On the other hand, end-system parallelism can be exploited to improve utilization of a single path by means of parallel streams or concurrent transfers. However, one should also consider system configuration since specific local constraints may introduce bottlenecks. One issue with all these techniques is that they cannot be ported to the clouds, since they strongly rely on the underlying network topology, unknown at the user-level.

Traditional techniques commonly found in scientific computing, e.g. relying on parallel file systems are not always adequate for processing big data on clouds. Such architectures usually assume high-performance communication between computation nodes and storage nodes. This assumption does not hold in current cloud architectures, which exhibit much higher latencies between compute and storage resources within a site, and even higher ones between datacenters.

3. Big Data and MapReduce Techniques

Big Data computing is an emerging data science paradigm of multi dimensional information mining for scientific discovery and business analytics over large scale infrastructure. The data collected/produced from several scientific explorations and business transactions often require tools to facilitate efficient data management, analysis, validation, visualization and dissemination, while preserving the intrinsic value of the data. The IDC report predicted that there could be an increase of the digital data by 40 times from 2012 to 2020. New advancements in semiconductor technologies are eventually leading to faster computing, large scale storage, faster and powerful networks at lower prices, enabling large volumes of data preservation and utilization at faster rate. Recent advancements in Cloud computing technologies are enabling to preserve, every bit of the gathered and processed data, based on subscription models, providing high availability of storage and computation at affordable price. Conventional data warehousing systems are based on pre-determined analytics over the abstracted data, and employs cleansing and transforming into another database known as data marts- which are periodically updated with the similar type of rolled-up data. However, Big Data systems work on non predetermined analytics; hence no need of data cleansing and transformations procedures.

Big Data organizes and extracts the valued information from the rapidly growing, large volumes, variety forms, and frequently changing data sets collected from multiple, and autonomous sources in the minimal possible time, using several statistical, and machine learning techniques. Big Data is characterized by 5V's such as Volume, Velocity, Variety, Veracity, and Value. Big Data and traditional data warehousing systems, however, have the similar goals to deliver business value through the analysis of data, but, they differ in the analytics methods and the organization of the data. In practice, data warehouses organize the data in the repository, by collecting it from other several databases like enterprise's financial systems, customer marketing systems, billing systems, point-of-sale systems, etc.

Businesses today are increasingly reliant on large-scale data analytics to make critical day-to-day business decisions. This shift towards data-driven decision making has fueled the development of MapReduce, a parallel programming model that has become synonymous with large scale, data-intensive computation. In MapReduce, a job is a collection of Map and Reduce tasks that can be scheduled concurrently on multiple machines, resulting in significant reduction in job running time. Many large companies, such as Google, Facebook and Yahoo!, routinely use MapReduce to process large volumes of data on a daily basis. Consequently, the performance and efficiency of MapReduce frameworks have become critical to the success of today's Internet companies.

A central component to a MapReduce system is its job scheduler. Its role is to create a schedule of Map and Reduce tasks, spanning one or more jobs, that minimizes job completion time and maximizes resource utilization. A schedule with too many concurrently running tasks on a single machine will result in heavy resource contention and long job completion time. Conversely, a schedule with too few concurrently running tasks on a single machine will cause the machine to have poor resource utilization. The job scheduling problem becomes significantly easier to solve if all map tasks have homogenous resource requirements in terms of CPU, memory, disk and network bandwidth. Indeed, current MapReduce systems, such as Hadoop Map- Reduce Version 1:x, make this assumption to simplify the scheduling problem. These systems use a simple slot-based resource allocation scheme, where physical resources on each machine are captured by the number of identical slots that can be assigned to tasks. Unfortunately, in practice, run-time resource consumption varies from task to task and from job to job. Several recent studies have reported that production workloads often have diverse utilization profiles and performance requirements. Failing to consider these job usage characteristics can potentially lead to inefficient job schedules with low resource utilization and long job execution time.

4. Problem Statement

Cloud computing environment provides scalable infrastructure for big data applications. Cross clouds are formed with the private cloud data resources and public cloud service components. Cross cloud service composition provides a concrete approach capable for large scale big data processing. Private clouds refuse to disclose all details of their service transaction records. History record based Service optimization method HireSome-II) is privacy aware cross cloud service composition method. QoS history records are used to estimate the cross cloud service composition plan. k-means algorithm is used as a data filtering tool to select representative history records. HireSome-II reduces the time complexity of cross cloud service composition plan for big data processing. The

following problems are identified from the current big data applications. They are Big data processing is not integrated with the system. Security and privacy for big data is not provided. Limited scalability in big data process. Mining operations are not integrated with the system.

5. Cloud Service Discovery and MapReduce Framework

Cloud computing environment provides resources to the users. Two types of resource sharing methods are provided in the cloud environment. They are public cloud and private cloud. The public cloud provides computational and service components to the users. Public cloud can be accessed by anyone. The private cloud is deployed to share resources to a group of users only. Big data sharing for the group of users are supported by the private cloud environment. Cross cloud framework integrates the public cloud resources with private cloud data values. The big data stored under the private cloud can be processed under the service components in the public cloud environment. Service components are selected with reference to the history records.

History record based Service optimization method HireSome-II) is enhanced to process big data values. Security and privacy is provided for cross cloud service composition based big data processing environment. Privacy preserved map reduce methods are adapted to support high scalability. The HireSome-II scheme is upgraded to support mining operations on big data. Data classification is carried out on the big data values with cross cloud resources. Navy bayesian classification algorithm is tuned to perform data classification with privacy ensured big data collections. Service components are called from the public cloud environment for the big data process.

The cloud computing supports large data sharing and computing environment. Public cloud environment provides resources for all users. The data values are provided under the private cloud storage environment. The hybrid cloud model integrates the public and private cloud resources. The service discovery operations are carried out with the history records. The History Record based Service Composition Method HireSome-II) scheme is adapted to search suitable service provides based on their performances. Aggregation based privacy model is used to protect privacy on service records. The K-means clustering algorithm is used to group the similar service calls. The HireSome-II scheme is integrated with the perform the mining operations on big data values. The privacy is provided for the big data values. The big data mining model uses the Thyroid data values. The classification operation is carried out to categorize the Thyroid patients with severity levels. K-anonymity scheme is employed to protect privacy on Thyroid data values. The Naive Bayes classifier is adapted for the data classification process. The MapReduce technique is also applied to select the resources for the data classification process.

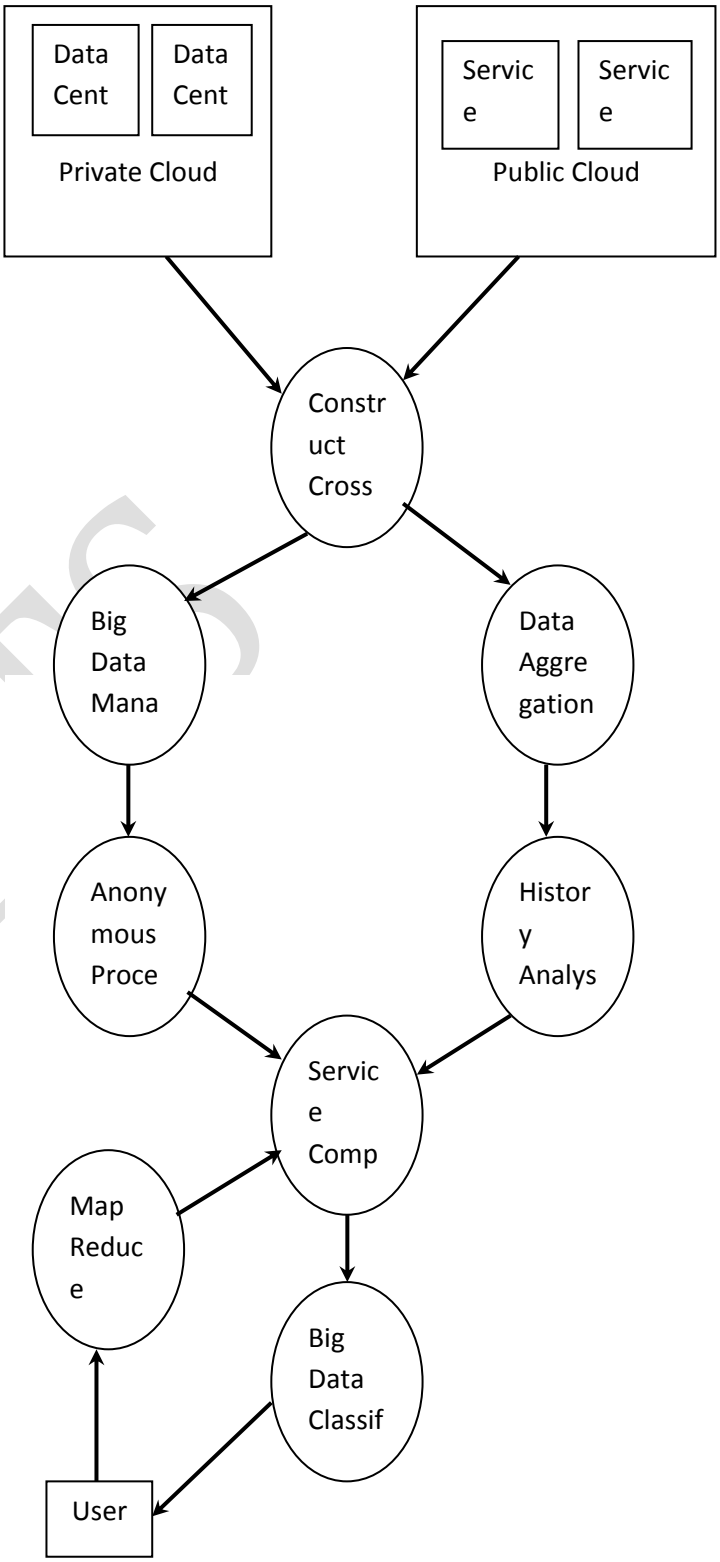

Figure No: 5.1. Cloud Service Discovery and MapReduce Framework

6. Experimental Analysis

The big data processing with privacy preservation scheme is constructed to perform big data classification and service composition tasks. The service composition methods are used to select the service providers with reference to their performance levels. Big data values are provided under the private cloud environment. Public cloud environment provides the service providers for the big data process. The public and private clouds are integrated to build the hybrid cloud or cross cloud framework. Service and data access information are maintained under the historical logs. The History record based Service optimization method HireSome-II) is adapted to handle the service provider selection process. Aggregation operations are used to protect the history records. K-Means clustering technique is used to group up the similar history records.

The History record based Service optimization method HireSome-II) is enhanced to perform big data processing with MapReduce techniques. Big data classification and service provider selection is carried out using the History record based Service Optimization method with Privacy Preserved MapReduce HireSome-II with PPMR) technique. The system is tested with different data count levels. Thyroid diagnosis data values are used in the system analysis process. The system is tested with three parameters. They are response time, throughput and failure rate values. The response time analysis is carried out to verify the service execution time analysis. Figure 6.1. shows the response analysis between the History record based Service optimization method HireSome-II) and the History record based Service optimization method with Privacy Preserved MapReduce HireSome-II with PPMR) scheme. The analysis result shows that History record based Service optimization method with Privacy Preserved MapReduce HireSome-II with PPMR) scheme reduces the response time 20% than the History record based Service optimization method HireSome-II) scheme.

The throughput analysis measures the data transfer rate between the user and the service providers. Figure 6.2. shows the throughput analysis between the History record based Service optimization method HireSome-II) and the History record based Service optimization method with Privacy Preserved MapReduce HireSome-II with PPMR) scheme. The analysis result shows that History record based Service optimization method with Privacy Preserved MapReduce HireSome-II with PPMR) scheme increases the throughput 15% than the History record based Service optimization method HireSome-II) scheme.

The failure rate analysis is carried out to compare the resource failure level under the service providers. Figure 6.3. shows the failure rate analysis between the History record based Service optimization method HireSome-II) and the History record based Service optimization method with Privacy Preserved MapReduce HireSome-II with PPMR) scheme. The analysis result shows that History record based Service optimization method with Privacy Preserved MapReduce HireSome-II with PPMR) scheme reduces the failure rate 10% than the History record based Service optimization method HireSome-II) scheme.

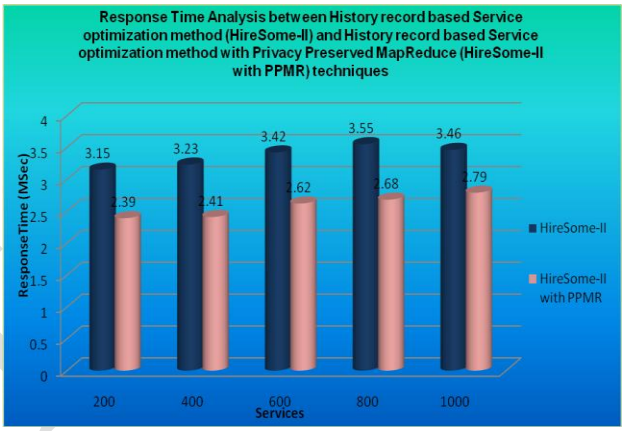

Figure No.6.1. Response Time Analysis between History record based Service optimization method (HireSome-II) and History record based Service optimization method with Privacy Preserved MapReduce (HireSome-II with PPMR) Techniques

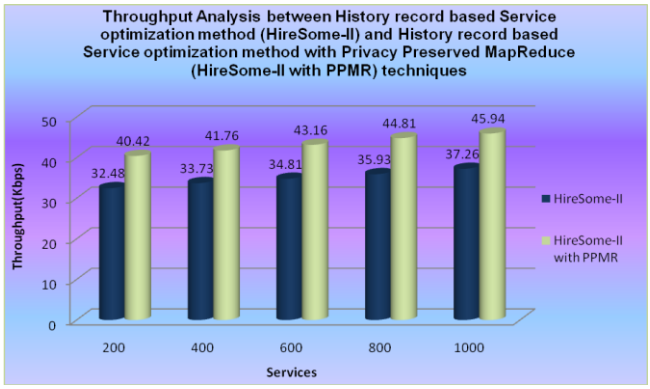

Figure No.6.2 Throughput Analysis between History record based Service optimization method (HireSome-II) and History record based Service optimization method with Privacy Preserved MapReduce (HireSome-II with PPMR) techniques

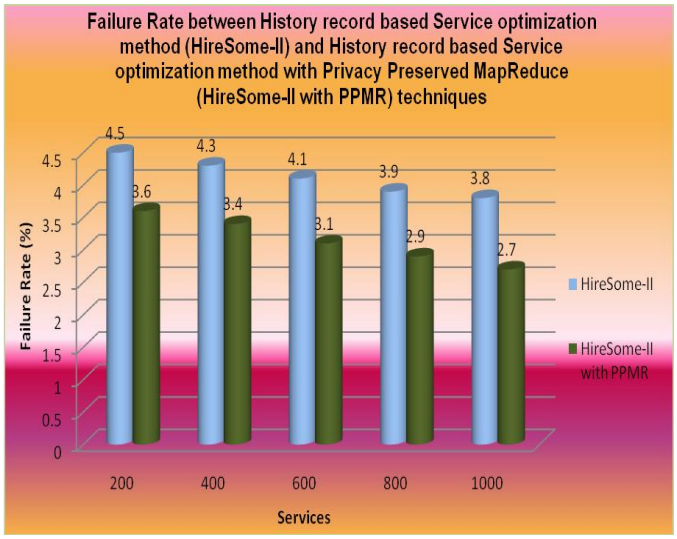

Figure No.6.3. Failure Rate between History record based Service optimization method (HireSome-II) and History record based Service optimization method with Privacy Preserved MapReduce (HireSome-II with PPMR) techniques

7. Conclusion and Future Work

The service composition scheme is designed for the cross cloud environment. Privacy preservation is applied on the service log data values. Clustering methods are applied to analyze the service log values. Resources are allocated using service composition methods. The system can be enhanced to support pattern based prediction process. The system can include the pricing factor for service discovery process.

References

1. T. Kosar, E. Arslan, B. Ross and B. Zhang, "Storkcloud: Data transfer scheduling and optimization as a service," in Proceedings of the 4th ACM Science Cloud '13, 2013, pp. 29–36.
2. "Cloudfront," http://aws.amazon.com/cloudfront/.
3. S. Pandey and R. Buyya, "Scheduling workflow applications based on multi-source parallel data retrieval," Comput. J., vol. 55, no. 11, pp. 1288–1308, Nov. 2012.
4. L. Ramakrishnan, C. Guok, K. Jackson, E. Kissel, D. M. Swany and D. Agarwal, "On-demand overlay networks for large scientific data transfers," in Proceedings of the 2010 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing, ser. CCGRID '10, 2010, pp. 359–367.
5. G. Khanna, U. Catalyurek, T. Kurc, R. Kettimuthu, P. Sadayappan, I. Foster and J. Saltz, "Using overlays for efficient data transfer over shared wide-area networks," in Proceedings of the 2008 ACM IEEE conference on Supercomputing, pp. 47:1–47:12.
6. G. Khanna, U. Catalyurek, T. Kurc, R. Kettimuthu, P. Sadayappan and J. Saltz, "A dynamic scheduling approach for coordinated wide-area data transfers using gridftp," in Parallel and Distributed Processing, 2008. IPDPS 2008., 2008, pp. 1–12.
7. T. J. Hacker, B. D. Noble and B. D. Athey, "Adaptive data block scheduling for parallel tcp streams," in Proc. of the 14th IEEE High Performance Distributed Computing, ser. HPDC '05, 2005, pp. 265–275.
8. W. Liu, B. Tieman, R. Kettimuthu and I. Foster, "A data transfer framework for large-scale science experiments," in Proceedings of the 19th ACM International Symposium on High Performance Distributed Computing, 2010, pp. 717–724.
9. C. Raiciu, C. Pluntke, S. Barre, A. Greenhalgh, D. Wischik, and M. Handley, "Data center networking with multipath tcp," in Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks, ser. Hotnets-IX, 2010, pp. 10:1–10:6.
10. W. Liu, B. Tieman, R. Kettimuthu and I. Foster, "A data transfer framework for large-scale science experiments," in Proceedings of the 19th ACM International Symposium on High Performance Distributed Computing, 2010, pp. 717–724.
11. R. L. Grossman, Y. Gu, M. Sabala and W. Zhang, "Compute and storage clouds using wide area high performance networks," Future Gener. Comput. Syst., vol. 25, pp. 179–183, 2009.
12. Radu Tudoran, Alexandru Costan and Gabriel Antoniu, "OverFlow-Multi-Site Aware Big Data Management for Scientific Workflows on Clouds", IEEE Transactions On Cloud Computing, Vol. X, No. X, August 2014
13. Suppawong Tuarob, Sumit Bhatia, Prasenjit Mitra and C. Lee, "AlgorithmSeer- A System for Extracting and Searching for Algorithms in Scholarly Big Data", IEEE Transactions On Big Data, 2016.