© 2016 IJSRST | Volume 2 | Issue 2 | Print ISSN: 2395-6011 | Online ISSN: 2395-602X
Themed Section: Science and Technology

# Secure and Trusted Information Brokering In Cloud Computing

**Jayalakshmi Kanagasabapathy*, C. Swaraj Paul**

Department of Computer Science and Engineering, Vels University, Chennai, Tamil Nadu, India

## ABSTRACT

To facilitate extensive collaborations, today's organizations raise increasing needs for information sharing via on-demand information access. Information Brokering System (IBS) a top a peer-to-peer overlay has been proposed to support information sharing among loosely federated data sources. It consists of diverse data servers and brokering components, which help client queries to locate the data servers. However, many existing IBSs adopt server side access control deployment and honest assumptions on brokers, and shed little attention on privacy of data and metadata stored and exchanged within the IBS. In this article, we study the problem of privacy protection in information brokering process. We first give a formal presentation of the threat models with a focus on two attacks: attribute-correlation attack and inference attack. Then, we propose a broker-coordinator overlay, as well as two schemes, automaton segmentation scheme and query segment encryption scheme, to share the secure query routing function among a set of brokering servers. With comprehensive analysis on privacy, end to- end performance, and scalability, we show that the proposed system can integrate security enforcement and query routing while preserving system-wide privacy with reasonable overhead. Finally, T-broker uses a lightweight feedback mechanism, which can effectively reduce networking risk and improve system efficiency. The experimental results show that, compared with the existing approaches, our T-broker yields very good results in many typical cases, and the proposed system is robust to deal with various numbers of dynamic service behavior from multiple cloud sites.

**Keywords:** Information Broking System, Automation segmentation, coordinates broker, privacy preserving, and Attribute-correlation attack

## I. INTRODUCTION

In recent years, we have observed an explosion of information shared among organizations in many realms ranging from business to government agencies. To facilitate efficient large-scale information sharing, many efforts have been devoted to reconcile data heterogeneity and provide interoperability across geographically distributed data sources.

In the context of sensitive data and autonomous data owners, a more practical and adaptable solution is to construct a data centric overlay including the data sources and a set of brokers helping to locate data sources for queries Such infrastructure builds up semantic-aware index mechanisms to route the queries based on their content, which allows users to submit queries without knowing data or server location. In our previous study, such a distributed system providing data access through a set of brokers is referred to as Information Brokering System (IBS).

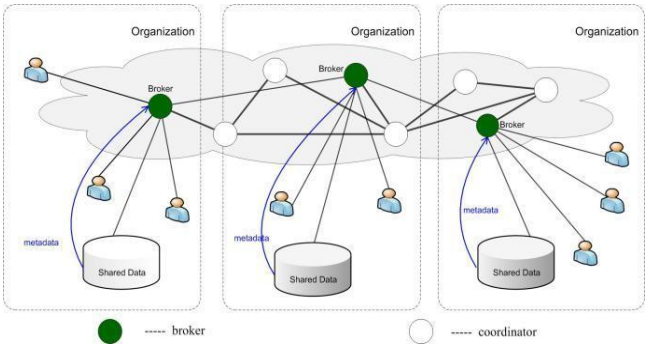


**Figure 1:** An overview of the IBS infrastructure.

While the IBS approach provides scalability and server autonomy, privacy concerns arise, as brokers are no

longer assumed fully trustable – they may be abused by insiders or compromised by outsiders. In this article, we present a general solution to the privacy-preserving information sharing problem. First, to address the need for privacy protection, we propose a novel IBS, named Privacy Preserving Information Brokering (PPIB). It is an overlay infrastructure consisting of two types of brokering components: brokers and coordinators. The brokers, acting as mix anonymizers, are mainly responsible for user authentication and query forwarding. The coordinators, concatenated in a tree structure, enforce access control and query routing based on the embedded nondeterministic finite automata – the query brokering automata. To prevent curious or corrupted coordinators from inferring private information, we design two novel schemes: (a) to segment the query brokering automata, and (b) to encrypt corresponding query segments. While providing full capability to enforce in-network access control and to route queries to the right data sources, these two schemes ensure that a curious or corrupted coordinator is not capable to collect enough information to infer privacy, such as "which data is being queried", "where certain data is located", or "what are the access control policies", etc. We show that PPIB provides comprehensive privacy protection for on-demand information brokering, with insignificant overhead and very good scalability.

### 1.1 Related Work

Research areas such as information integration, peer-to-peer file sharing systems and publish-subscribe systems provide partial solutions to the problem of large scale data sharing. Information integration approaches focus on providing an integrated view over large numbers of heterogeneous data sources by exploiting the semantic relationship between schemas of different sources. The PPIB study assumes that a global schema exists within the consortium, therefore, information integration is out of our scope. Peer-to-peer systems are designed to share files and data sets (e.g. in collaborative science applications). Distributed hash table technology is adopted to locate replicas based on keyword queries. The coarse granularity (e.g. files and documents) still makes them short of our expressiveness needs. Further, P2P systems may not provide complete set of answers to a request while we need to locate all relevant data.

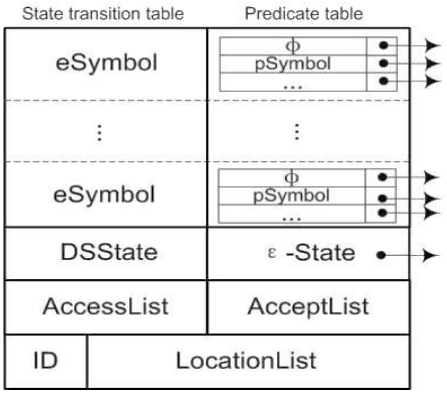


**Figure 2 :** Data structure of an NFA state.

### 1.2 Vulnerabilities and the Threat Model

In a typical information brokering scenario, there are three types of stakeholders, namely data owners, data providers, and data requestors. Each stakeholder has its own privacy:

(1) The privacy of a data owner (e.g. a patient in RHIO) is identifiable data and the information carried by this data (e.g. medical records). Data owners usually sign strict privacy agreements with data providers to protect their privacy from unauthorized disclosure/use.

(2) Data providers store collected data, and create two types of metadata, namely routing metadata and access control metadata, for data brokering. Both types of metadata are considered privacy of a data provider. (3) Data requestors disclose identifiable and private information in the querying process. For example, a query about AIDS treatment reveals the (possible) disease of the requestor.

We adopt the semi-honest (i.e., honest-but-curious) assumption for the brokers, and assume two types of adversaries, outside attackers and curious or corrupted brokering components. Outside attackers passively eaves drop communication channels. Curious or corrupted brokering components follow the protocols properly to fulfill their functions, while trying their best to infer others' private information from the information disclosed in the querying process

### 1.3 Attribute-correlation attack

An attacker intercepts a query (in plaintext), which typically contains several predicates. Each predicate

describes a condition, which sometimes involves sensitive and private data (e.g. name, SSN or credit card number, etc.). If a query has multiple predicates or composite predicate expressions, the attacker can "correlate" the corresponding attributes to infer sensitive information about the data owner. This attack is known as the attribute correlation attack:

Example 1. A tourist Diana is sent to the emergency room at California Hospital. Doctor Bob queries for her medical records through a medicare IBS. Since Diana has the symptom of leukemia, the query has two predicates: [name="Diana"], and [symptom="leukemia"]. Any malicious broker that has helped routing the query could guess "Diana has a blood cancer" by correlating two predicates in the query.

## II. METHODS AND MATERIAL

### 2. Previous Implementation

### 2.1 Privacy-Preserving Query Brokering Scheme

While QBroker seamlessly integrates the content-based indexing function into the NFA-based access control mechanism, it heavily relies on the QBroker for the enforcement and shifts all the data (i.e., the ACR, index rules, and user queries) to it. However, if the QBroker is compromised or no longer assumed fully trusted (e.g. under the honest-but-curious assumption as in our study), the privacy of both the requestor and the data owner is under risk. To tackle the problem, we present a privacy-preserving information brokering (PPIB) infrastructure with two core schemes. The automata segmentation scheme divides the QBroker into multiple logically independent components so that each component only needs to process a piece of an user query but still can fulfill the original brokering functions via collaboration. The query segment encryption scheme allows to encrypt query pieces with different keys so that one automaton component can decrypt the responsible piece(s) for further processing, while not hurdling the original distributed indexing function. The existing brokering architecture for cloud computing do not consider user feedback only relying on some direct monitoring information.

There is no doubt that the efficiency of a trust system is an important requirement for multiple cloud environments. That is, the trust brokering system should be fast convergence and light-weight to serve for a large number of users and providers. However, existing studies paid little attention to this question, which greatly affects scalability and availability of the trust system.

#### 2.1.1 Automaton Segmentation

In the context of distributed information brokering, multiple organizations join a consortium and agree to share the data within the consortium. While different organizations may have different schemas, we assume a global schema exists by aligning and merging the local schemas. Thus, the access control rules and index rules for all the organizations can be crafted following the same shared schema and captured by a global automaton, the global QBroker. The key idea of the automaton segmentation scheme is to logically divide the global automaton into multiple independent yet connected segments, and physically distribute the segments onto different brokering servers.

**Segmentation:** The atomic unit in the segmentation is an NFA state of the original automaton. Each segment is allowed to hold one or several NFA states. We further define the granularity level to denote the greatest distance between any two NFA states contained in one segment. Given a granularity level k, for each segmentation, the next i 2 [1; k] NFA states will be divided into one segment with a probability 1=k. Obviously, a larger granularity level indicates that each segment contains more NFA states, resulting in a smaller number of segments and less end-to-end overhead in distributed query processing. On the contrary, a coarse partition is more likely to increase the privacy risk. The tradeoff between the processing complexity and the privacy requirements should be considered in deciding the granularity level. As privacy protection is of the primary concern of this work, we suggest a granularity level

1) To reserve the logical connection between the segments after segmentation, we define heuristic segmentation rules:
2) Multiple NFA states in the same segment should be connected via parent-child links;

3) No sibling NFA states should not be put in the same segment without the parent state; and
4) The "accept state" of the original global automaton should be put in separate segments. To ensure the segments are logically connected,
5) We change the last states of each segment to be "dummy" accept states, which point to the segments holding the child states in the original global automaton.

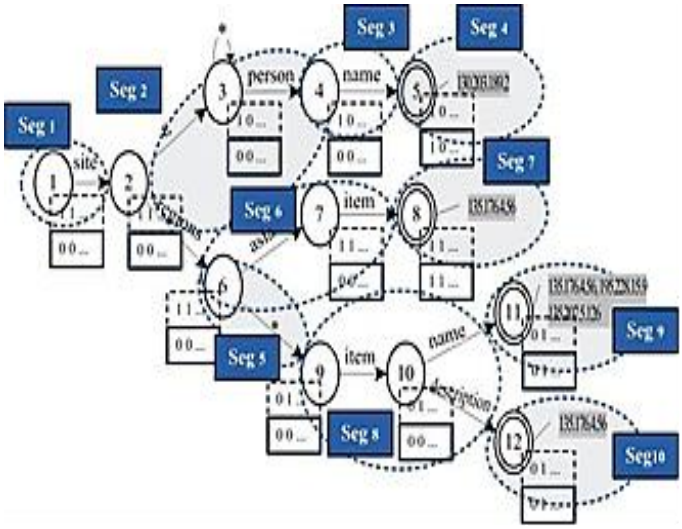


**Figure 3 :** Divide the global automaton with granularity

## 3. System Implementation

As mentioned above, most current cloud brokering systems do not provide trust management capabilities to make trust decisions, which will greatly hinder the development of cloud computing. depicts the brokering scenario in existing and Aeolus We can see that this existing brokering architecture for cloud computing do not consider user feedback only relying on some direct monitoring information. As depicted T-broker architecture, a service brokering system is proposed based on direct monitoring information and indirect feedbacks for the multiple cloud environments, in which T-broker is designed as the TTP for cloud trust management and resource matching. Before introducing the principles for assessing, representing and computing trust, we first present the basic architecture of T-broker and a brief description of its internal components.

### 3.1 Sensor-Based Service Monitoring (SSM)

This module is used to monitor the real-time service data of allocated resources in order to guarantee the SLA (Service Level Agreement) with the users. In the interactive process, this module dynamically monitors the service parameters and is responsible for getting run-time service data. The monitored data is stored in the evidence base, which is maintained by the broker. To calculating QoS-based trustworthiness of a resource we mainly focus on five Kinds of trusted attributes of cloud services, which consist of node spec profile, average resource usage information, average response time, average task success ratio, and the number of malicious access.

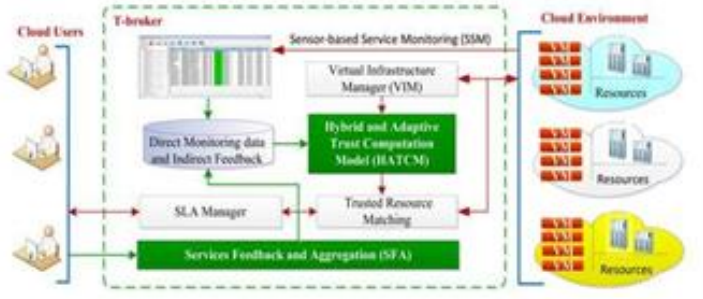


**Figure 4 :** Proposed Architecture

The node spec profile includes four trusted evidences: CPU frequency, memory size, hard disk capacity and network bandwidth. The average resource usage information consists of the current CPU utilization rate, current memory utilization rate, current harddisk utilization rate and current bandwidth utilization rate. The number of malicious access includes the number of illegal connections and the times of scanning sensitive ports.

### 3.2 Proposed Model

The proposed system is robust to deal with various numbers of dynamic service behavior from multiple cloud sites. Some hybrid trust models are proposed for cloud computing environment It is no doubt that how to adaptively fuse direct trust (first-hand trust) and indirect trust (users' feedback) should be an important problem, however, most current studies in hybrid trust models either ignore the problem or using subjective or manual methods to assign weight to this two trust factors (first-hand trust and users' feedback).

The proposed trust management framework for a multi-cloud environment is based on the proposed trust evaluation model and the trust propagation network. First, a trusted third party-based service brokering architecture is proposed for multiple cloud environments, in which the T-broker acts as a middleware for cloud trust management and service matching. T-broker uses a hybrid and adaptive trust model to compute the overall trust degree of service resources, in which trust is defined as a fusion evaluation result from adaptively combining the direct monitored evidence with the social feedback of the service resources.

### 3.2.1 Cloud User Module

Cloud users can send request to the T-broker for accessing the cloud resources, the feedback system collects locally-generated users' ratings and aggregates these ratings to yield the global evaluation scores. After a user completes a transaction, the user will provide his or her rating as a reference for other users in future transactions.

### 3.2.2 Cloud Resources Module (Admin)

Cloud resource module will provide the cloud resources. web based cloud computing managing tool for managing cloud infrastructure from multiple providers. Right Scale enables organizations to easily deploy and manage business-critical applications across public, private, and hybrid clouds. Spot Cloud provides a structured cloud capacity marketplace where service providers sell the extra capacity they have and the buyers can take advantage of cheap rates selecting the best service provider at each moment. a cloud is modeled in seven layers: Facility, network, hardware, OS, middle ware, application, and the user. These layers can be controlled by either the cloud provider or the cloud customer.

### 3.2.3 T-Broker Module

In this module T-broker uses some sub modules,

**(1)Trust-aware brokering architecture**

In which the broker itself acts as the TTP for trust management and resource scheduling. Through distributed soft-sensors, this brokering architecture can real-time monitor both dynamic service behavior of resource providers and feedbacks from users.

**(2)Hybrid and Adaptive Trust Computation Model (HATCM)**

A hybrid and adaptive trust model to compute the overall trust degree of service resources, in which trust is defined as a fusion evaluation result from adaptively combining dynamic service behavior with the social feedback of the service resources. The HATCM allows cloud users to specify their requirements and opinions when accessing the trust score of cloud providers. That is, users can specify their own preferences, according to their business policy and requirements, to get a customized trust value of the cloud providers

**(3) Maximizing deviation method (MDM)**

A maximizing deviation method to compute the direct trust of service resource, which can overcome the limitations of traditional trust models, in which the trusted attributes are weighted manually or subjectively. At the same time, this method has a faster convergence than other existing approaches.

**(4) Sensor-Based Service Monitoring (SSM)**

This module is used to monitor the real-time service data of allocated resources in+ order to guarantee the SLA (Service Level Agreement) with the users. In the interactive process, this module dynamically monitors the service parameters and is responsible for getting run-time service data. The monitored data is stored in the evidence base, which is maintained by the broker. To calculating QoS-based trustworthiness of a resource we mainly focus on five kinds of trusted attributes of cloud services, which consists of node spec profile, average resource usage information, average response time, average task success ratio, and the number of malicious access.

**(5)Virtual Infrastructure Manager (VIM)**

Each cloud provider offers several VM configurations, often referred to as instance types. An instance type is defined in terms of hardware metrics such as CPU

frequency, memory size, hard disk capacity, etc. In this work, the VIM component is based on the OpenNebula virtual infrastructure manager this module is used to collect and index all these resources information from multiple cloud providers. It obtains the information from each particular cloud provider and acts as a resource management interface for monitoring system. Cloud providers register their resource information through the VIM module to be able to act as sellers in a multi-cloud marketplace. This component is also responsible for the deployment of each VM in the selected cloud as specified by the VM template, as well as for the management of the VM life-cycle. The VIM caters for user interaction with the virtual infrastructure by making the respective IP addresses of the infrastructure components available to the user once it has deployed all VMs.

**(6)Service level agreement Manager (SLA)**

In the multiple cloud computing environment, SLA can offer an appropriate guarantee for the service of quality of resource providers, and it serves as the foundation for the expected level of service between the users and the providers An SLA is a contract agreed between a user and a provider which defines a series of service quality characters. Adding trust mechanism into the SLA management cloud brokering system can prepare the best trustworthiness resources for each service request in advance, and allocate the best resources to users.

### 3.3 Multiple Clouds Computing:

MULTIPLE cloud theories and technologies are the hot directions in the cloud computing industry, which a lot of companies and government are putting much concern to make sure that they have benefited from this new innovation However, compared with traditional networks, multiple cloud computing environment has manyunique features such as resources belonging to each cloud provider, and such resources being completely distributed, heterogeneous, and totally virtualized; these features indicate that unmodified traditional trust mechanisms can no longer be used in multiple cloud computing environments. A lack of trust between cloud users and providers has hindered the universal acceptance of clouds as outsourced computing services.

### 3.4 Feedback Aggregation:

The "Trust as a Service" (TaaS) framework to improve ways on trust management in cloud environments. In particular, the authors introduce an adaptive credibility model that distinguishes between credible trust feedbacks and malicious feedbacks by considering cloud service consumers' capability and majority consensus of their feedbacks. However, this framework does not allow to assess trustworthiness based on monitoring information as well as users' feedback. In large-scale distributed systems, such as grid computing, P2P computing, wireless sensor networks, and so on, feedback provides an efficient and effective way to build a socialevaluation based trust relationship among network entities. By the same token, feedback also cans provider important reference in evaluating cloud resource trustworthiness. Consider large-scale cloud collaborative computing environment which host hundreds of machines and handles thousands of requests per second, the delay induced by trust system can be one big problem. So, there is no doubt that the computational efficiency of a feedback aggregating mechanism is the most fundamental requirement. As depicted in Fig. 3, we build cloud social evaluation system using feedback technology among virtualized data centers and distributed cloud users, and we use a lightweight feedback mechanism, which can effectively reduce networking risk and improve system efficiency.

**Table 1:** Service of Behaviours

| Trust attributes | QoS indicators (service behavior) |
|---|---|
| node spec profiles | CPU frequency<br>memory size<br>hard disk capacity<br>network bandwidth |
| average resource usage information | current CPU utilization rate<br>current memory utilization rate<br>current hard disk utilization rate<br>current bandwidth utilization rate |
| average response time | average response time |
| average task success ratio | average task success ratio |
| the number of malicious access | the number of illegal connections<br>the times of scanning sensitive ports |

## III. RESULT AND DISCUSSION

### 4.1 Accuracy Evaluation

The value of εri (λ) is used to measure the degree of deviation of calculating results; thus, the closer its value is to zero, the higher the calculating accuracy. First, observing MAD under conditions with different number of training samples(Note: we gather a training sample dt = (dt1, dt2, . . . , dtm) at each time-stamp t, so the number of training samples equals to the number of time-stamps). In order to observe experimental results under different scale of training samples, we use two kinds of inputting samples, a small number of training samples and a large number of training samples.

In the first group of experiments, the total number of training samples changes from 10 to 50. shows that the number of training samples has a direct effect on the accuracy of the trust models. When the number of training samples is small (t < 30), the MADs of three models are more than 0.20. When the number of training samples is set larger (t ≥ 30), the MADs of the other two models are more than 0.23. The MAD of our trust model is less than 0.20, the MAD of our trust model is much smaller than that of STM and FTM, which reflects that our model's performance is better than that of other models under conditions with different number of training samples.

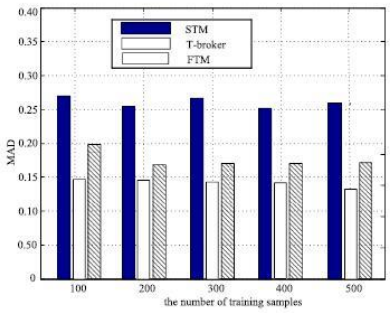


**Figure 5:** The values of MAD with a large number of training samples.

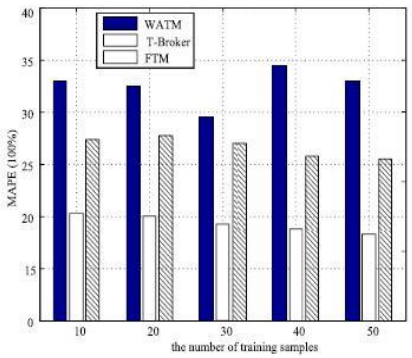


**Figure 6 :** The values of MAPE with a small number of training samples.

The MAPE is a measure of accuracy in a fitted time series value in statistics, specifically trending. It usually expresses accuracy as a percentage. MAPE can reflect the unbiasedness of the calculating model. A smaller value of MAPE reflects the calculating model has better and unbiased accuracy. We also use two kinds of inputting samples to evaluate the MAPE of the three models, a small number of training samples and a large number of training samples.

## IV. CONCLUSION

In this paper, we present T-broker, a trust-aware service brokering system for efficient matching multiple cloud services to satisfy various user requests. Experimental results show that T-broker yields very good results in many typical cases, and the proposed mechanism is robust to deal with various number of service resources. In the future, we will continue our research from two aspects. First is how to accurately calculate the trust value of resources with only few monitored evidences reports and how to motivate more users to submit their feedback to the trust measurement engine. Implementing and evaluating the proposed mechanism in a large-scale multiple cloud system, such as distributed data sharing and remote computing, is another important direction for future research.

## V. REFERENCES


[1] Hamid Sadeghi (2011), "Empirical Challenges and solutions in constructing a high-performance metasearch engine", emeraldinsight.

[2] C.Swaraj Paul , G. Gunasekaran, "A Descriptive Literature Survey on Search of Data inCloud " in International Journal of Applied Engineering Research IJAER, pp. 13112-13114, Volume 10, Number 17 (2015) Special Issues, ISSN 0973-4562.

[3] Leonidas Akritidis, Dimitrios Katsaros ∗, Panayiotis Bozanis (2011), "Effective rank aggregation for metasearching", The Journal of Systems and Software 84 (2011) 130–143

[4] Craswell, N. and Hawking, D. (2002), "Overview of the TREC-2002 web track", Proceedings of the 11th Text Retrieval Conference(TREC), National Institute of Standards and Technology, Gaithersburg, MD, pp.86-95.

[5] Dwork, C., Kumar, R., Naor, M., Sivakumar, D., 2001. Rank aggregation methods for the Web. In: Proceedings of the ACM International Conference on World Wide Web (WWW), pp. 613–622.

[6] Farah, M., Vanderpooten, D., 2007. An outranking approach for rank aggregation in information retrieval. In: Proceedings of the ACM International Conference on Research and Development in Information Retrieval (SIGIR).

[7] H. Kim, H. Lee, W. Kim, and Y. Kim, "A trust evaluation model for QoS guarantee in cloud systems," Int. J. Grid Distrib. Comput., vol. 3, no. 1, pp. 1–10, Mar. 2010.

[8] Renda, M.E., Straccia, U., 2003. Web metasearch: rank vs score based rank aggregation methods. In: Proceedings of the ACM International Symposium on Applied Computing (SAC), pp. 841–846.

[9] C.Swaraj Paul , G. Gunasekaran, "An Optimized Attribute Based Similarity Search In Metric Database " in International Journal of Applied Engineering Research IJAER, pp. 15631-15641, Volume 10, Number6 (2015) Special Issues, ISSN 0973-4562.

[10] Vogt, C.C., Cottrell, G.W., 1999. Fusion via a linear combination of scores. Information Retrieval 1(3), 151-173.

[11] Aslam, J.A., Montague, M.H., 2001a. Metasearch consistency. In: Proceedings of the ACM International Conference on Research and Development in Information Retrieval (SIGIR), pp. 386–387.

[12] P. Jain, D. Rane, and S. Patidar, "A novel cloud bursting brokerage and aggregation (CBBA) algorithm for multi cloud environment," in Proc. 2nd Int. Conf. Adv. Comput. Commun. Technol. (ACCT), Jan. 2012, pp. 383–387.