International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 9, No.5 (Sep-2020)

http://dx.doi.org/10.12785/ijcds/090508

Service-Oriented Broker for Effective Provisioning of Cloud Services – a Survey

Rajganesh Nagarajan[1] and Ramkumar Thirunavukarasu[2]

[1] Department of Information Technology, A.V.C. College of Engineering, Mayiladuthurai, India
[2]School of Information Technology & Engineering, Vellore Institute of Technology, Vellore, India

Received 3 Aug. 2019, Revised 11 May 2020, Accepted 25 Jul. 2020, Published 1 Sep. 2020

Abstract: Services computing plays a vital role in the field of information technology and enables users to perform web and cloud services in more efficient and effective manner. In the life cycle of services computing, phases such as service discovery, composition, and delivery of services and managing services as per the Service Level Agreements (SLA) have been received significant attention by the research community. Though web services and cloud services are two instances of services computing, their user communities significantly differ in various aspects, which include service specification, consumption, and adherence to service agreements. This paper approaches services computing from the perspective of two architectural paradigms namely Service-oriented Architecture (SOA) and Cloud computing. The existing research attempts performed in the phases of service discovery, composition and provisioning of services as per the SLA has been extensively reviewed from the perspective of SOA and Cloud. Based on the literature review, a number of research issues are also summarized towards achieving service excellence further. At the end, the paper emphasizes the need for service-oriented broker to enhance the discovery and provisioning process of cloud services.

Keywords: Services Computing, Service Discovery, Service Composition, Service Provisioning, Service-Oriented Broker.

1. INTRODUCTION

Services computing [1] refers to a flexible computing paradigm to develop platform-independent, autonomous software application termed as 'services. These services communicate with each other by passing data in a well-defined, acceptable format, or by coordinating an activity between services. Services computing offer two major paradigms namely 'Service-oriented Architecture (SOA)' and 'Cloud computing' for supporting the life cycle of services. Both allow users to perform activities such as service creation, discovery and composition, delivery of services. The revolution of cloud computing allows the users to access the services more efficiently and effectively through internet and cloud environments.

A. Service-oriented Architecture (SOA)

To begin with, this section explores the underlying concept of SOA followed by service models of cloud computing. Service-oriented architecture [2] offers loosely coupled software services that are independent in nature and can be accessed without the knowledge of underlying platform. A typical SOA architecture [3], which involves three major players namely service producer, service consumer, and service registry is shown in Fig.1. In SOA paradigm, the registry acts like a middleware between the service provider and consumer. It exposes various capabilities of service producer and offers re-usable business services as a solution to the commonly re-occurring problems on time. Therefore, SOA services are considered as horizontal one since it focuses single domain namely 'Business Services'.

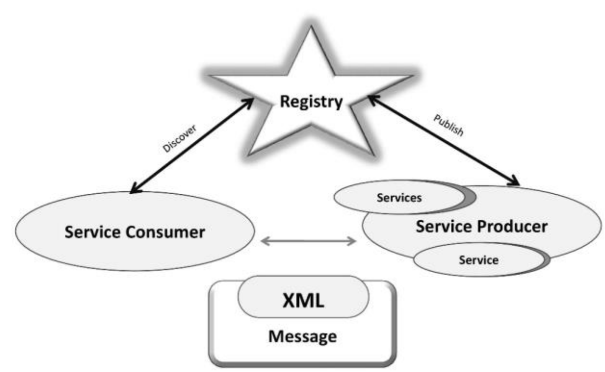

Figure 1. A typical service-oriented architecture model

E-mail:anuvarajganesh@gmail.com, ramkumar.thirunavukarasu@vit.ac.in

B. Cloud Computing

Another computing environment that offers on-demand, pay-per-use, and utility-oriented services to the end user is called as 'cloud computing' [4]. With the advent of cloud computing, users can avail their required services without holding their own infrastructure [5]. In cloud computing, a service is any kind of resources that can be provided over the internet. Hence, cloud computing offers various levels of services such as 'Infrastructure', 'Platform', and 'Software' to the consumer [6]. This layered level of service architecture makes the services as a vertical one. In 'Infrastructure as a Service (IaaS)' model, the users can avail the infrastructure-based services such as physical computers, storage, network resources as per their requirement, and they are allowed to configure the availed resources as per their wish. In general, the 'IaaS' service can be obtained either as public, private or as a combination of these two. The model of 'Platform of Service (PaaS)' offers the computing platform that includes operating systems, program developing environment, web server and database to the cloud users. These platform level resources can be shared and re-used by the user community as per their computational needs. Similarly, the 'Software as a Service (SaaS)' model delivers the application services with pre-defined compound functionalities through the internet. Fig. 2 depicts the services offered by a cloud computing model.

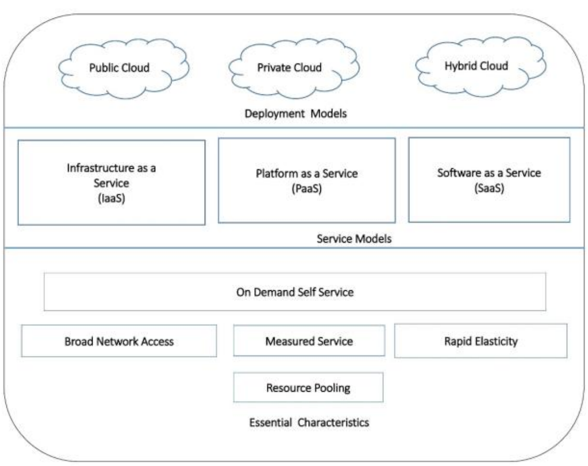

Figure 2. Cloud computing model – NIST Reference

The architectural style of SOA and cloud computing is unique while offering services to the users. In a nutshell, SOA is an architecture, whereas cloud computing is an instance of the architecture with the inclusion of certain value-added features. SOA aims to offer business solution by creating, organizing, reusable software components, while cloud computing provides set of allowable technologies with the capability of serving a bigger and more convenient platform for building solution over the internet. In SOA, web services are consumed as services of re-usable nature, wherein cloud computing, services are consumed through 'IaaS', 'PaaS', 'SaaS' in a broad manner. By excluding the core concept such as on-demand provisioning, utility computing, virtualization, one can conclude that the architecture of SOA provides a model for establishing cloud-based services. In general, cloud computing is closely understood as a form of web services that are offered by traditional Service-oriented Architecture (SOA).

C. Difficulties of Normal Cloud Broker

In both the models (SOA and Cloud) [7], the service consumer and provider are present, they differ from each other with the existence of service registry (in SOA), cloud broker (in Cloud). In this perspective, both the entities have been considered as middleware for the provisioning of services. The service registry in SOA simply offers a fine-grained service namely web service discovery and such kind of discovery process are not enough for the cloud environment since services are vertical in nature. To cater this, cloud computing includes an entity, which termed as broker [8] that has been designed to serve as an umpire between cloud consumer and provider. Also, it has the following limitations.

(i) In some circumstances, the cloud broker fails to update itself with respect to technical requirements of consumers that would result the selection of in-appropriate services.

(ii) The poor understanding or reasoning capability of the cloud broker may lead to recommend un-wanted service details to the cloud user and henceforth failed to fulfill the exact requirements of user community.

(iii) Due to the security policies and constraints, sometimes brokers would augment complexities while processing services requests.

From the perspective of cloud service provider, the broker must help the user to avail services according to their budgetary levels and should create new opportunities towards growth in service sales. Hence, it is more appropriate to extend the functionalities of cloud broker with the inclusion of computational intelligence aspects that are most suitable for cloud computing paradigm. Since the origin of cloud paradigm emerged from the concept of services computing, we approach our proposed broker as service-oriented broker that would minimize the hazels involved in the phases of cloud life cycle by performing various value-added functionalities.

D. Objective and Contribution of the Review

Though various literature surveys have been contributed in the broad area of services computing, the main objective of this paper is to perform systematic cum comprehensive review of research works that have been carried out with the aid of service registry and cloud broker found in the SOA and Cloud computing respectively. Further, the paper attempts to design a cloud broker framework with self-healing properties based on the notion of service registry found in SOA for efficient selection and provisioning of cloud services. Hence, this paper is a new kind of survey in the arena of services computing.

Accordingly, the main contributions of the paper are:

i) A detailed survey with respect to the phases of services computing (discovery, composition, and delivery) is carried out from the perspectives of SOA and Cloud.
ii) The research issues in the existing phases of services computing have been addressed and a detailed analysis is performed.
iii) Emphasizes the need of service-oriented broker for the cloud computing with a proposed architecture.
iv) Possible solutions are highlighted with the novel technique that remains open for future research.

The paper has been organized as follows: Section 2 discusses the state-of-the-art research works with respect to SOA and Cloud. In Section 3, the related research works in the service composition phase is addressed. Similarly, Section 4 highlights the provisioning of services with Service Level Agreement. The Emerging research issues and the need for Service-oriented broker are deliberated in Section 5. In section 6, a service-oriented broker for the cloud computing framework is proposed with its functional components. Finally, Section 7 concludes the paper and suggests possible solutions for future works.

2. PHASE OF SERVICE DISCOVERY

Web services are the form of SOA implementation to support interoperable machine-to-machine interaction over a network. This interoperability is gained through a set of XML-based open standards, such as WSDL (Web Service Definition Language), SOAP (Simple Object Access Protocol), and UDDI (Universal Description, Discovery, and Integration). These standards provide a common approach for defining, publishing, and using web services. In the context of SOA, service discovery is the process of locating web service providers, and retrieving web services descriptions that have been previously published. A service registry contains relevant metadata about available and upcoming services as well as pointers to the corresponding service contract documents that can include SLAs. Service discovery in cloud computing refers to find suitable services on the basis of available service description. A service description contains functional, non-functional capabilities along with the characteristic of services. The discovery is constrained by functional, technical specifications and budgetary requirements along with the appropriate security policies concerned.

A. SOA based Service Discovery

Sycara et al. [9] approached web service discovery in terms of OWL-S enabled brokers that replaced the matchmaking principles adopted earlier. The matchmaker [10, 11] maps offered services against the consumer's requirements and encourages in identifying the provisions of the producer. Here, the matchmaker acts like a service descriptor for identifying the capabilities of service producers. When further service enhancement arises, the consumer can select the services of matchmaker again. With the inclusion of broker concept, the process of service discovery has been regularized and the broker is responsible not only for identifying the services, but also regulates the services between the consumer and producer. However, the architecture fails to make an interaction with the multi-agents for gathering the service information.

Crasso et al. [12] developed a system called Web Service Query by Example (WSQE) for simplifying the web service discovery. Their intention in their work is to define the service information in terms of its descriptions. Hence, the authors are allowed to avail the services even without knowing the complete details. Rajendran et al. [13] developed an algorithm to perform service matching, ranking, and selection. Here, the user's preferences are considered for ranking of services. However, the method fails to evaluate the overall quality from the calculated weights.

Yager et al. [14] proposed a method to identify the best desirable services that fit the user's requirements by the lexico-graphical preferences. A single threshold value has been used to represent the consumer's satisfaction level with the boundary values of acceptable and unacceptable attributes. With this stated principle, the inconsistency may arise due to improper decision of the user. In addition, the authors have not adapted a valid method to decide the weight and threshold objectively.

De Souza and Rabelo [15] performed the dynamic discovery of services and offered an integrated, comprehensive service environment. It considered functional and non-functional requirements apart from business contexts [16]. If any of the required service is not available, the discovery operation has been initiated to find out the suitable services. However, the dynamic nature of services discovery may lead to selection of inappropriate services at some time.

Garcia and Sim [17] proposed an agent-based approach to compose services in multi-cloud environments for different types of services. The developed agents autonomously and successfully dealt with changing service requirements through self-organization and collaboration. From that, the authors reduced the search space needed for discovery mechanisms and obtained the exact service description with respect to the consumer requirements. Incorporation of functional and non-functional requirements while performing filtering operations would be one of the further enhancements to their work.

Zhang et al. [18] extended their work with Support Vector Machine (SVM) technique for the categorization of services in the repositories. The proposed model looks like a black box for increasing the service categorization as per domain knowledge. The authors achieved a user-centered service categorization and enrichment of domain ontology. Nevertheless, the proposed SVM technique failed to support the categorization of multiple domains and hence further enhancement is needed for repository-oriented service search engine.

Mistry et al. [19] attempted to develop an architecture for the improvement of SOA and performed semantic discovery of web services with the inclusion of actors named Alignmentor and Ontology handler. The actors of existing SOA framework improved the matching process and generated the service ontology for efficient discovery of services. Peng [20] applied the K-means clustering algorithm for generating feature vector-based service description. Clustering is one promising method for unsupervised data. However, their proposal failed to include more classification methodologies for finding the best match.

Choi and Jeong [21] proposed a trust-based system to calculate the priority information of the attributes (cost, fault rate, response time, operability, privacy, and availability) stated by the consumer. After that, pairwise comparison has been carried out for ranking purpose. The performance of this work may be improved by incorporating the semantic concepts while discovering the services. In addition, the contextual information about the services plays a vital role in identifying the appropriate services. Hence, the incorporation of contextual details certainly increases effectiveness of the service discovery process.

Ramacher and Monch [22] proposed a hierarchical approach to perform the composite service selection. Here, the authors proposed an integrated solution for a cost-minimizing tactical service selection. It has been shown that the proposed approach works for service compositions of a realistic size and the tactical, operational objectives are achieved in an environment with uncertain QoS attributes.

Zhang et al [23] helped the service requesters to obtain relevant services by exploiting domain knowledge about service functionalities (i.e., service goals) mined from textual descriptions of services. They have extracted service goals from services' textual descriptions using an NLP-based method and clustered service goals by measuring their semantic similarities. However, the non-functional properties are not considered and hence the integrated service discovery is not achieved.

B. Cloud Based Service Discovery

Parhi et al. [24, 25] proposed a multi-agent-based framework for the discovery and selection of cloud services. The work used the domain depended ontology for ensuring a standard vocabulary for specifying cloud service descriptions by the cloud service providers and at the same time indicate the cloud user about their service preferences. This framework processes the user request at different stages according to the stated functional and non-functional requirements and constructs the semantic cloud ontology for the service selection process. However, the monitoring and controlling of different agents are not addressed in this work. Hence, quality of the final service selection and the process effectiveness are not guaranteed.

Nabli et al. [26] performed the service discovery with cloud service ontology. The cloud ontology is used to calculate the service similarities for their selection. The construction of ontology with the semantic network leads to increase the complexity of service representation. Similarly, the proposal has not addressed the functional and non-functional requirements. Hence, the need for the Natural Language Queries (NLQ) for the retrieval of cloud service information is needed to improve service discovery.

Nagarajan et al. [27] proposed a semantic network-based technique to perform the service selection. This work constructed a cloud ontology before the service selection process. With the aid of semantic network, appropriate services are identified by performing an intersection search. Similarly, the increase in number of cloud services also poses significant challenges in the selection of required cloud services. To resolve such issues, Nawaz et al. [28] proposed a cloud broker architecture for cloud service selection by finding the pattern of changing priorities of user preferences through Markov chain. However, the model improves the cloud service selection process; the complexity in finding user prioritization about the services is also increased.

Kumar et al. [29] introduced a framework for determining the most suitable candidate cloud services by integrating the Analytical Hierarchical Process (AHP) and Technique for Order Preference by Similarity to Ideal Solution (TOPSIS). The Pairwise comparison method is used in the service similarity calculation and then the ranking of the services are achieved through TOPSIS method. While considering the huge collection of services, the pairwise comparison is not suitable for the identification of similar services.

Sun et al. [30] identified the disadvantages of linear MCDM methods while performing service selection and proposed a solution to avoid the manual service ranking. Their work considered the nonlinear preferences based on criteria interactions. Hence, the service selection is achieved even without having enough historical information. However, this model is not suitable to process the newly offered services, which are not having its QoS ratings.

Al-Faifi et al. [31] proposed a hybrid Multi Criteria Decision Method (MCDM) to evaluate and rank cloud service providers from smart data. The hybrid method applied the k-means algorithm to consolidate cloud service providers with similar features and ranked the clusters using Analytical Network Process (ANP). Furthermore, this method considered the interdependencies and relations between the performance measurements. However, the weighted method for clustering and ranking leads to the performance degradation.

Nie et al. [32] used Analytic Hierarchy Process (AHP) to calculate weights for factors such as security, cost, and reputation to perform service discovery. However, the proposed approach failed to include QoS related criteria in its hierarchy for decision-making and unable to consider the variability of QoS. In addition, the authors are not focused the monitoring process for the effective discovery of cloud service.

Ding et al. [33] proposed an innovative ranking prediction approach for personalized cloud service selection. Instead of complicated service evaluation process, they used Collaborative Filtering (CF) algorithm for the service selection. However, the trustworthiness of the services has not addressed properly.

Mezni and Abdeljaoued [34] developed a Fuzzy Formal Concept Analysis to generate reliable service selection using lattice representation. The fuzzy lattice representation discarded the poorly rated cloud services and provided accurate service collections. However, the time required for the identification of suitable cloud services increased the complexity. In addition, parsing the large concept lattices depends on the number of cloud services and the degree of changes in their properties and ratings.

Nagarajan et al. [35, 36] elaborated the cloud service selection with the aid of fuzzy logic and MapReduce framework. A set of predefined QoS factors are considered for deriving the trust level of services before the selection. In another work, Nagarajan and Thirunavukarasu [37] explored their views with respect to the uncertainty in service specification process. As a solution, a fuzzy based model is developed to guide the inexperienced user during their service selection process.

Modi and Garg [38] proposed a framework for cloud service matchmaking, selection and composition based on the Semantic Web and QoS parameters such as, availability, response time, throughput and cost. The authors attempted to integrate different cloud service providers into a multi-cloud environment. Due to this, the overall complexity of the proposed system is increased. Further, the interoperability of different services from multi-cloud environment is not addressed properly.

Pang et al. [39] proposed a behavior based service-oriented process model in cloud environment. Here, the authors built a global service composition model based on semantic relations between property concepts and a service composition planning method. Through this, the discovery of cloud service is improved and assured service trustworthiness too.

Ma et al. [40] developed a model to address the quality issues while selecting the services. A variation-aware approach is proposed to select an optimal cloud service according to users' non-functional requirements. In addition, the proposed work utilized the TOPSIS method by considering both the objective QoS variation and subjective user preferences during different time periods for the prediction of service similarity for selection. However, handling the large collection of services is not addressed in this work.

Wang et al. [41] proposed a cloud service selection model based on the user preferences about the trust value of services. Therefore, this proposal finds out the similarity of service attribute preferences, scores and evaluation similarities. In addition, the improved hierarchical clustering algorithm is used to cluster in view of forming user preference domain. Based on this user grouping, the service trust has been measured for the selection. However, the identity and the authenticity of the user in the group are not considered in their work.

Qian et al. [42] developed a model for discovering the IaaS type of services from the cloud service providers. By considering the geographical location of the user, better service discovery has been achieved. However, the author has not ensured the reliability of the service. As a solution, Mastroianni and Papuzzo [43] improved the effectiveness of service discovery with clustered service descriptors based on its frequency value. Through this, the co-occurrences of services, response time, bandwidth, and processing load are significantly reduced.

Zhygmanovskyi and Yoshida [44] presented a detailed study in service provisioning. They proposed a P2P technology to perform the discovery and sharing of services with Distributed Hash Table (DHT) approach. This approach allows sliding doors between the service descriptions and executed the queries. Tang et al. [45] proposed a method to assess the service trustworthiness by using the user feedback ratings. The objective and subjective trust assessment values are aggregated to assess the overall trustworthiness of services. But, user feedback on trustworthiness has not been accounted in their work.

3. PHASE OF SERVICE COMPOSITION

Service composition [46, 47] is a design principle of service-orientation paradigm, which persuades and reuses the services from multiple vendors. In SOA, user can get the service details from the registry and access the specified services from the respective producer. When multiple vendors are providing the same service, selecting an optimal one at par with different Quality of Service (QoS) attributes also causes significant problem in the phase of service composition.

In cloud computing, the resources such as software, hardware, and network are treated as services and effectively composed in an on-demand fashion. With the concept of virtualization in cloud computing, the users can avail their services without the intervention of cloud providers. However, handling of dynamic service requirements is always a challenging task in the case of cloud-based service composition. Research works that are pertain to service composition phase of SOA and cloud are discussed in the following sub-sections.

A. SOA based Service Composition

Blake [48] introduced the concept of reasoning mechanism for accessing exact service descriptions. The descriptions about the services are created with the help of 'Web Services Description Language (WSDL)'. To enhance the message communication between the service provider and consumer, the author used 'Simple Object Access Protocol (SOAP)', a lightweight protocol for exchanging the services. Hence, the author treated both the description of services and protocol, a broker based one.

Wang et al. [49] attempted a model with the incorporation of an agent for providing design and run-time composition of services. In design level, the web services are summarized as an agent model at the implementation level, UDDI, WSDL, and SOAP provide discovery, descriptions, and communications for service compositions respectively [50, 51]. Their work broadly explained about the ontology workflow and its coordination in the web services composition.

Chen & Paik [52] developed a service composition approach by advocating Global Social Service Network (GSSN) in view of assisting higher-quality service composition. The QoS attributes are inter-linked to create relationships among them and the reduction of search space has been achieved by applying investigation among the services. The authors proposed a quality-driven workflow-search algorithm for the identification of quality links to provide the services with minimal cost to the end users. However, the proposed system failed to perform the dynamic adaptation of customer preferences and hence lacks in account customer's feedback.

Rostami et al. [53] used OWL-S language for the description of web services. In their work, they used clustering techniques for the categorization of web services. They used ant-colony algorithm for finding the best set of web services that have high degree of combining ability. From the proposed system, the authors achieved web service composition with different challenging constraints within an optimal period.

Sheng et al. [54] performed the web service composition by describing a sequence of activities such as service orchestration and service choreography. In service orchestration, a single process is responsible for coordinating the various services. The choreography is related to the interactions of multiple services in the environment and symbolizes the participatory services. The authors also categorized the service composition in two aspects namely static and dynamic. The static composition dealt with the aggregation of services at design time, whereas the later one allows the replacement of services at runtime.

Li et al. [55] discussed different types of web service composition approaches along with their limitations. In addition, they proposed an evidence-based method for validating the service proposal before the composition. Hence, the authors performed the service composition based on SOA implementation, which decreases the complexity and difficulty in availing web services.

Ghobaei-Arani et al. [56] addressed the problem of web service composition in geo-distributed cloud environments. They proposed a cuckoo search algorithm, to solve the web service composition problem, which considers not only the QoS of the web services but also the network QoS such bandwidth, latency, delivery, and availability. The dynamic QoS prediction is not focused and hence new services without the QoS are discarded.

B. Cloud based Service Composition

In cloud computing, Zou et al. [57] proposed a framework for service composition in multi-cloud base environments. The authors applied an AI based planning and combinatorial optimization process to perform composition of cloud services. The grouping of cloud service provider according to the cloud user requirement is done with set covering model. The model organizes the cloud services in the form of tree like structure and used an approximation algorithm (AI based) to enhance the service selection and service composition in a multi-cloud environment.

Gavvala et al. [58] used an eagle strategy for designing QoS aware cloud service composition. In this approach, the exploration is done similar to how an eagle searches for its prey initially. Once the prey is found, the eagle changes its behavior for hunting the prey. Likewise, exploration applied optimization technique to complete the task. By using this, a balance between exploration and exploitation is achieved to overcome the issues like slow convergence rate or premature convergence. Further, they considered that all the required services are available with a single service-repository and hence the integration of multiple services has not been achieved in their work.

Naseri and Navimipour [59] proposed an agent-based method to compose services by identifying the QoS parameters. This work employed the Particle Swarm Optimization (PSO) algorithm to select the best services based on fitness function. Similarly, Merizig et al. [60] proposed an agent-based architecture with a new cooperation protocol, which offered an automatic and adaptable service composition by providing a composite service with the maximum QoS. The approaches improved the service composition with better QoS, but failed to ensure the correctness of service QoS values.

To overcome such issues, Garcia and Sim [61] designed an agent to compose services from the federated cloud. The proposal aims to reorganize the consumer's requirements (fees, functional and non-functional specifications) for best provisioning of services [62]. Accordingly, their work attempted to introduce individual agents for representing the services with ontology and assisted both service provider, and consumer with the help of broker agents. With this distributed nature of different agents, the authors provided an un-interrupted service composition.

In addition to that, Wei and Blake [63] developed a system with Check Period Relaxation (CPR) and Modified-CPR (MCPR) adaptive algorithms to control the interactions between the agents. However, their work did not provide better services from the operational cloud environment. Hence, the issue of agent's interaction with acknowledgment needs to be focused.

4. Provisioning of Services With Service Level Agreement

In SOA, the contract-based service provisioning has not been focused well, whereas cloud computing offers these features through Service Level Agreements (SLA). Hence, the section focus towards SLA based cloud service provisioning.

In a cloud environment, consumers are varying with their requirements and cloud providers are offering different service capabilities based on their marketing strategies. Therefore, identifying the policies for fulfilling the customer's expectations is an important issue and prompts the need for a valid agreement [64]. A service level agreement [65, 66] is a formal agreement, which helps to identify cloud consumers' expectations, clarify responsibilities, and facilitate communication between providers and consumers. Cloud consumers need proper agreements before the migration of their infrastructure to cloud data center and, the providers need SLA to ensure the delivery of quality services to end-users [67]. It is difficult to satisfy the consumer's need from the perspective of service providers and thereof an optimal strategy is to be arrived through the process called 'Negotiation' [68]. Various scenarios of negotiation are to be considered for the cloud environment [69]. The first one is the straight negotiation between the involved parties. Here, the providers are creating a standard template with stated criteria's (duration, charge and response). In the second scenario, a trusted agent can participate on behalf of consumers to define critical parameters of SLA. The third scenario supports more than one agent in the process of negotiation. These types of negotiation would be an effective strategy if the consumer requires multiple services from various vendors.

Wei et al. [70] presented a method to schedule the available services with mutual quality of service requirements. They have suggested the notion of game theory to resolve the crisis in resource allotment. The method solves the independent optimization problem through the binary programming. Alhamad et al. [71, 72] proposed architecture for SLA management in a cloud environment. They contributed an agent-based framework for SLA negotiation and monitoring of services. Accordingly, the consumer is signed in the SLA contract before using the services. Niehorster et al. [73] proposed a most common 'SaaS' architecture portal for easy web access. Consumers can define their needs with QoS factors for availing the services. After the cost estimation process, the agent imposes the fulfillment of the stated Service Level Objectives (SLO) and minimizes the resources. Their work directs the agents towards the providers, not on themselves. Therefore, there is no such possibility to interact among the available agents for provisioning of resources.

Buyya et al. [74] estimated the service consumption through the agents. Hence, they ensured the resource scheduling process without any SLA breaches. However, the schemes for negotiating SLAs have not been focused well in their work. Wieder et al. [75] introduced SLA management (business policies, SLA templates, negotiation, provisioning, and adjustments) for service-oriented cloud environments. Torkashvan and Haghighi [76] proposed an approach to define parameters such as quality, availability, and reliability using XML language. However, the negotiation process has not expressed in their work. Rao et al. [77] proposed their work with the aid of game-theoretic approach in the service provisioning phase of cloud computing. They have accounted a strategy model for the effective functioning of the cloud infrastructure.

Al Falasi et al. [78] classified the research efforts on SLA into two types such as SLA management and Cloud-specific SLA. The SLA management has been developed for a specific cloud provider and for the federated one. Through this, SLA establishment for both individual and federated cloud environment are achieved. The Cloud-specific SLA (CSLA) is another model for cloud services provided by Nie et al. [79], based on WSLA proposed by Keller and Ludwig [80]. In their proposal, the agents are handled multiple requirements' definition into one aggregated SLA document. Once SLA gets agreed and service provisioning is commenced. Here a management model provides a means to deploy SLA, measures the performance, evaluates the SLA and billing the service. This model did not provide a clear specification of SLA negotiation and did not consider the dynamic nature of SLAs in cloud environments.

Wu et al. [81] used the resource provisioning approach to reduce the cost and avoided the violations occurred during the SLA process. The authors proposed an algorithm to minimize the occurrences of penalty and performed the rescheduling of services to fulfill the cloud user. Due to the repetition process, the waiting time of the customer got significantly affected. Anisetti et al. [82] proposed e-auctions for service selection and ranking before the SLA process. This approach simplified the selection process by taking the service bids that are partially matched.

Chen et al. [83] proposed a cloud broker framework based on dynamic game theory to achieve successful negotiation between the participators. In addition, the authors proposed a 'Nash equilibrium point' and a 'satisfaction degree' to find out the optimal value of SLA attributes such as price and bandwidth. Messina et al. [84] proposed an intelligent agent-based protocol to compare and understand different service provider's SLA. At first, the protocol has focused to perform negotiation with respect to technical information of services. It has been designed such a way to focus the understanding of semantic and technical terms such as availability, quality, security, and defect rates of the required services.

The following Table [Table I] summarizes the various research efforts in Service-oriented Architecture (SOA) and Cloud computing with respect to service discovery, composition, and provisioning of services.

5. Emerging Research Issues and the Need For Service-oriented Broker

The various extensive research attempts discussed above explored various emerging issues in the phases of cloud life cycle. The issues are tailored and the need for service- oriented broker in the cloud-computing framework has been elaborated in this section.

A. Emerging issues in the Cloud Life Cycle Phases

In service discovery, certain challenges such as handling the incorrect details, uncertainties in requirements specification, dynamic discovery of service, and user/item-based service discovery are considered for the discussion.

i) ***Elimination of incorrect service details:*** The identification of right service [17] from the service repository is always a challenging one for the new cloud user. The nature of the service must be classified and grouped together to form the service group [21]. Followed that, the classification of services must be performed and the functionalities are to be recorded with the broker for simplifying the service discovery process. Similarly, during the requirement specification, the user may face the problem in understanding the service details. This scenario is termed as uncertainties [23, 24, 25] and should be addressed properly by adopting the fuzzy logic-based principle named 'linguistic variables' [32, 33].

ii) ***Dynamic service discovery:*** Though many literatures [23, 24, 25, 26, 27, 38] expressed the capabilities and action list of the broker in service discovery, the challenges are still existing while performing dynamic discovery of services. To cater this, a service-oriented broker must be constructed to perform tasks such as pre-processing, ranking, and for construction of ontology towards facilitating automatic service discovery. Whenever a new service is published, the broker must evaluate them against the user's requirement on the basis of soft computing benchmarks such as Fuzzy Logic and machine learning principles such as Artificial Neural Networks and Decision Tree.

iii) ***User or service based discovery:*** The user or service-based discovery [85] is an ongoing research issue in services computing. In user-based discovery, the identification and grouping of similar users with their service interest is not always be stable. A user may purchase a service for the usage of another user; he may be entirely different from the identified group. Hence, it is appropriate to consider contextual and personalized information about the users while availing the service discovery process.

After the service identification, the integration of required services from the federated cloud environment in an important process in the composition phase. The following issues are to be addressed properly in order to accumulate the services in an effective manner.

i) ***Interaction among the multiple cloud providers:*** A service pack with the collection of all possible services from the federated cloud environment is an important one for the effective handling of cloud services. Normally, the cloud computing offers services from the providers like Amazon, Google, and Salesforce.com etc. The integration of services from the available providers with respect to the service nature is not yet achieved. Hence, the need for the service-oriented broker in the existing cloud computing is prompted to enhance the service provisioning capabilities.

ii) ***Effective handling of the trusted parties:*** Utilization of services from the trusted third parties improves the customer satisfaction. Hence, the need to identify the trusted parties and their integration is a promising research avenue.

TABLE I. SUMMARY OF RESEARCH EFFORTS IN SOA AND CLOUD COMPUTING

S. No	Contributor(s)	Key Highlights	Identified Gap	Service Discovery		Service Composition		SLA based Service Provisioning		Existence of Broker/ Agent
				SOA	Cloud	SOA	Cloud	SOA	Cloud	
1	Sycara et al., 2004	Matchmaking principles for service discovery	Interaction among multi-agents	✓	-	✓	-	-	-	✓
2	Crasso et al., 2008	Service descriptions are developed	Difficult to represent huge collection of service details.	✓	-	-	-	-	-	-
3	Rajendran et al., 2010	Perform service-matching, ranking based on user preferences.	Poor evaluation of overall quality from the calculated weights	✓	-	-	-	-	-	✓
4	Yager et al., 2011	User behavior-based selection	Limitations of service details weight and threshold values.	✓	-	-	-	-	-	-
5	De Souza and Rabelo, 2011	Dynamic discovery of services	Inappropriate service selection	-	✓	-	-	-	-	-
6	Garcia et al., 2013	Search space reduction is achieved	Functional and Non-functional requirements are not considered.	✓	-	-	-	-	-	-
7	Zhang et al., 2012	User-centered service categorization	Failed to support the categorization of multiple domains.	-	✓	-	-	-	-	-
8	Mistry et al., 2012	Semantic discovery of web services	Difficult to represent the ontology	✓	-	-	-	-	-	-
9	Peng, 2012	Clustering based discovery	Poor classification methodologies	✓	-	-	-	-	-	-
10	Choi and Jeong, 2014	Trust based framework for service discovery	Service's QoS values are missing	✓	-	-	-	-	-	✓
11	Ramacher and Monch, 2015	Hierarchical approach for composite service selection	QoS values are not considered	✓	-	✓	-	-	-	-
12	Zhang et al, 2018	NLP-based discovery	Non-functional properties of services are not considered	✓	-	-	-	-	-	-
13	Parhi et al., 2018	Multi-agent with ontology-based framework	Poor coordination among Agents.	-	✓	-	-	-	-	-
14	Nabli et al., 2018	Semantic Network and Ontology based method	Natural language queries cannot be processed.	-	✓	-	-	-	-	-
15	Nagarajan et al., 2017	Semantic network and cloud ontology based	Searching time is increased	-	✓	-	-	-	-	✓
16	Nawaz et al., 2018	Markov chain with user preferences	Complexity in finding user prioritization	-	✓	-	-	-	-	-

17	Kumar et al., 2018	AHP and TOPSIS based method	Poor performance of pairwise comparison with huge services	-	✓	-	-	-	-	-
18	Sun et al., 2019	Non-linear preferences are considered to avoid the issues of normal MCDM	Missing to predict the newly arrived services.	-	✓	-	-	-	-	-
19	Al-Faifi et al., 2019	Hybrid method applied k-means to find out the similar services and ranked using ANP	Performance degradation due to weighted method	-	✓	-	-	-	-	-
20	Nie et al., 2011	AHP based method	Changes in the QoS values are not considered	-	✓	-	-	-	-	-
21	Ding et al., 2017	Personalized cloud service selection - CF based.	Service trustworthiness is not ensured	-	✓	-	-	-	-	-
22	Mezni and Abdeljaoued, 2018	Poor rated services are ignored to provide better service collection	Time complexity gets increased	-	✓	-	-	-	-	-
23	Nagarajan et al., 2018	Fuzzy with MapReduce Framework	Need to overcome the uncertainty issues	-	✓	-	-	-	-	✓
24	Nagarajan and Thirunavukarasu, 2019	Fuzzy Decision Tree based representation to overcome the uncertainties	Dynamic prediction of service's QoS is missing	-	✓	-	-	-	-	✓
25	Modi and Garg, 2019	Services from multi cloud environment is attempted	Interoperability between multiple services is not achieved	-	✓	-	-	-	-	-
26	Pang et al., 2019	Service trustworthiness is achieved	Not suitable for multi-cloud environment	-	✓	-	-	-	-	-
27	Ma et al., 2019	QoS based service selection	Poor performance with large collection of services.	-	✓	-	-	-	-	-
28	Wang et al., 2019	User preferences-based service selection	Malicious user information is not detected	-	✓	-	-	-	-	-
29	Qian et al., 2013	Location based service discovery – user context	Poor service reliability	-	✓	-	-	-	-	-
30	Mastroianni and Papuzzo, 2014	Improved service discovery	Suggestion of mismatched services	-	✓	-	-	-	-	-
31	Zhygmanovskyi and Yoshida, 2014	Distributed Hash Table based approach	Difficult to process large volume of services	-	✓	-	-	-	✓	-
32	Tang et al., 2017	User feedback based	Malicious users are not restricted	-	✓	-	-	-	-	-

		trustworthy prediction								
33	Blake, 2003	Ontology based service composition	Inferencing is very difficult	-	-	✓	-	-	-	✓
34	Wang et al., 2006	Run time service composition is performed	Difficult to analyze the ontology	-	-	✓	-	-	-	✓
35	Chen and Paik, 2015	Service's QoS are considered in composition	Failed to adapt the customer feedbacks	-	-	✓	-	-	-	-
36	Rostami et al., 2014	Ant-colony based method for service composition	Clustering failed to compose versatile service collections	-	-	✓	-	-	-	-
37	Sheng et al., 2014	Static and dynamic service composition is achieved	Need of centralized control	-	-	✓	-	-	-	-
38	Li et al., 2012	Evidence based service composition	No proper evaluation of service trust	-	-	✓	-	-	-	-
39	Ghobaei-Arani et al., 2017	QoS based service composition	Failed to predict the new services QoS	-	-	✓	-	-	-	-
40	Zou et al., 2010	Multi cloud service composition is achieved	Increased the overall cost of the selection process	-	-	-	✓	-	-	-
41	Gavvala et al., 2019	QoS based service composition	Integration of multiple cloud services are not possible	-	✓	-	✓	-	-	-
42	Naseri and Navimipour, 2019	Agent based QoS service composition	Poor prediction of Service's QoS	-	✓	-	✓	-	-	-
43	Merizig et al., 2018	Agent based QoS service composition (automatic)	Poor prediction of Service's QoS	-	✓	-	✓	-	-	-
44	Garcia and Sim, 2010	Services composition from federated cloud	Difficult to co-ordinate the agents	-	✓	-	✓	-	-	-
45	Wei and Blake, 2013	Agent based QoS service composition	Not provided the better services from the operational cloud environment	-	-	-	✓	-	-	-
46	Wei et al., 2010	Game theory applied for service allotment	Binary programming not suitable for multi- cloud environment	-	-	-	✓	-	✓	-
47	Alhamad et al., 2010	SLA based service provisioning	Co-ordination among the agents are not considered	-	-	-	-	-	✓	-
48	Niehorster et al., 2010	Agents based service provisioning	Poor interaction among the agents	-	-	-	-	-	✓	✓
49	Buyya et al., 2011	Agents based service provisioning	Negotiation part of SLA needs to be improved	-	-	-	-	-	✓	-
50	Wieder et al., 2011	SLA based service provisioning	Understanding between provider and user is more critical	-	-	-	-	-	✓	-

51	Torkashvan and Haghighi, 2012	Agreements based service provisioning	SLA needs to be improved	-	-	-	-	-	✓	-
52	Rao et al., 2012	Game-theoretic approach used for service provisioning	SLA part is missing	-	-	-	✓	-	✓	-
53	Al Falasi et al., 2013	Service provisioning with SLA	No clear specification of SLA negotiation	-	-	-	-	-	✓	-
54	Nie et al., 2012	Service provisioning with SLA	Not considered the dynamic nature of SLA	-	-	-	-	-	✓	-
55	Keller and Ludwig, 2003	Service provisioning with SLA	Not considered the dynamic nature of SLA	-	-	-	-	-	✓	-
56	Wu et al., 2014	Minimized the penalties in service provisioning	Response time is affected due to rescheduling of available services	-	-	-	-	-	✓	-
57	Anisetti et al., 2014	Improved service provision with SLA	Partial service details affected the expected result	-	✓	-	✓	-	✓	-
58	Chen et al., 2016	Better negotiation framework for service provisioning	Uncertainties are not considered	-	-	-	-	-	-	✓
59	Messina et al., 2016	Intelligent agent-based service composition	Not sure about the service's QoS	-	-	-	-	-	✓	✓

iii) ***Legitimate budgetary model for effective service handling:*** Based on the budgetary constraints proposed by the cloud users, the categorization of services must be performed. It can be accomplished by the techniques such as the adaptation of optimal cost estimation model using the genetic algorithm for resource utilization, inclusion of multi-criteria decision-making techniques for designing attribute matrices towards budgetary plans, and delivering traceability metrics for effective mapping between posted requirements and stated budget.

iv) ***Issues related to service migration on the cloud:*** In existing cloud computing framework, the operating principles are motivated towards normal discovery and provisioning of services from a single provider. Therefore, the existing methodologies are failed to compose the needed services from more than one cloud provider. With the introduction of the interoperability nature, the migration of services from one cloud service provider to another can be achieved.

Finally, service delivery with qualitative service list and the proper agreement between the provider and consumer is performed. The responsibility of this phase is to evaluate the service quality, preparation of agreements, handling the customer issues and updating the service details with respect to the dynamic changes in the services computing. Various research efforts on service level agreements and QoS have been considered by many researchers in services computing. In web service, 'WS-Agreement' and 'Web Service Level Agreement Language and Framework' are used for describing SLA. These are developed from Open Grid Forum (OGF) and IBM respectively. In cloud computing, new standards are to be derived while designing a broker since SLA matrices adapted in the web services are not well enough for the cloud.

From the literature, we have listed some of the issues that could be addressed to improve the service provisioning through the broker.

i) ***Evaluation of service quality:*** Certifying the services with respect to the customer requirements is still an existing problem in the cloud computing. One should consider the factor of service credibility in terms of quality of service factors (availability, reliability, accessibility, pricing factor) before recommending them to the user.

ii) ***Dynamic adaptation of customer feedback:*** A feedback gateway for recording the user response about the offered services is exist now. However, the analysis of the feedbacks against its authentication becomes a major issue in all sorts of services computing. Hence, an idea to validate the reality of the cloud user should be accounted for the future scope.

iii) ***Privacy issues related to customer details:*** Though privacy emerged as a major issue in services computing, it must be properly treated while designing the broker. At any levels of functionalities, the user interface or business logic of the proposed broker should not try to acquire the personal information of the customer. Hence, it is necessary to incorporate the various service metrics for handling the privacy issues.

iv) ***Service up-gradation alerts:*** According to consumer-provider agreement, the contract among the entities are transient in nature and it ends once the service life cycle has been completed. A broker that offers service up-gradation alert even after the completion of the contract would be a novel choice by more consumers [86, 87]. It can be done with the aid of feature selection theory, rough set theory for implementing automated service up-gradation alerts by the broker.

B. The Need for Service-oriented Broker

Though the cloud service model is an extension of Service-oriented Architecture, cloud users are availing various value-added vertical services as per their dynamic requirements. Hence, the layer of service brokerage is to be included among cloud users and consumers. The service brokerage layer would enhance the core cloud functionalities such as service discovery, service integration, service aggregation, service customization, quality assurance, and service optimization. The intelligence in brokerage can be instilled through the service-oriented broker. A service-oriented broker can be defined as software entity that provide a platform of service brokerage for both cloud users and providers and offer computationally intelligent cloud services through artificial intelligence, soft computing, machine learning and data analytical techniques for achieving service excellence in the phases of cloud life cycle.

6. SERVICE-ORIENTED BROKER FRAMEWORK FOR CLOUD COMPUTING

This section proposes a cloud system with the inclusion of service-oriented broker for effective provisioning of services. A typical cloud model with the inclusion of service-oriented broker is shown in Fig.3. The functional components of the proposed framework are explained here.

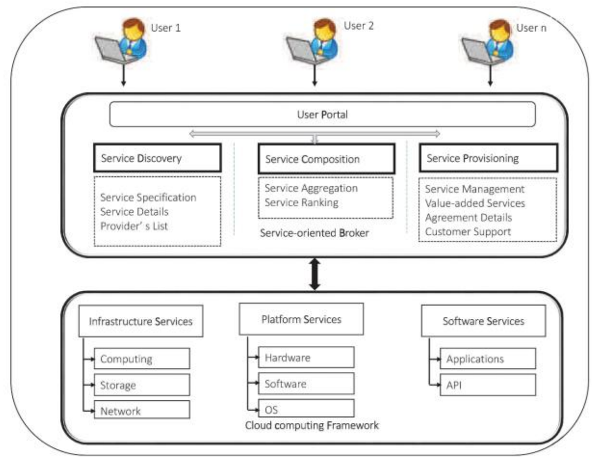

Figure 3. An idea for the future cloud-computing framework with service-oriented broker

(i) User portal for requirements gathering: This module has been proposed towards the fulfillment of user expectations with respect to the cloud services. As per the inclusion of our service-oriented broker, this module can analyze the requirement based on expected cost and features. As shown in Table II, a typical end-user portal accepts the IaaS specific input parameters such as computing, storage, and network details along with on-demand time (duration) and expected cost (budget).

Consider a typical scenario where, three cloud users 'CU1', 'CU2', and 'CU3' approaching our broker with the following IaaS requirements: the user 'CU1' needs the requirement of 2.5 GHz speed of computing, 8 GB of RAM, 1 TB of hard disk space with 4 Gbps of network for the duration of 60 minutes with the stated cost of $0.25.

TABLE 2 END-USER PORTAL FOR INPUT SPECIFICATIONS

Cloud User (CU)	IaaS type Services				Cost per Hour
	Cpu Instance (GHz)	Storage		Network Bandwidth (Gbps)	
		RAM (GB)	HDD (TB)		
CU1	2.5	8	1	4	$ 0.25
CU2	1.4	4	1	1	$ 0.15
CU3	3.4	16	2	8	$ 0.30

(ii) Service discovery and composition: This module focuses the intelligent part of the proposed broker. Accordingly, the service selection task has been carried out in an intelligent way. The broker applies various sort of intelligent techniques for the effective service selection. As an example, we put forward a service-oriented broker based on the concept of fuzzy ontology to rank the discovered services. The broker considers the functional, non-functional characteristic, security policies of the services with the budgetary constraints and constructs an ontological structure with appropriate weights. With respect to our example, broker lists the possible cloud services along with the providers which are very closer to the posted requirements of user CU1 (Table 3).

TABLE 3 SELECTED SERVICES [FOR CU1]

Cloud User (CU)	IaaS type Services				Cost per Hour
	Cpu Instance (GHz)	Storage		Network Bandwidth (Gbps)	
		RAM (GB)	HDD (TB)		
CU1	2.5	8	1	4	$ 0.25
CU2	1.4	4	1	1	$ 0.15
CU3	3.4	16	2	8	$ 0.30

(iii) Service Provisioning: Finally, the broker prepares a Service Level Agreement (SLA) for the binding of the consumer with a provider. The SLA includes aspects such as service scope, quality, and responsibilities of both provider and consumer. The SLA establishment through the proposed service-oriented broker always aims to assure the attributes such as customer support, the trustworthiness of service, long time availability and guaranteed service updates for the enhancement of provisioned services. In addition, the proposed broker periodically investigates the services and creates a service log to ensure the service effectiveness and their business levels. Such self-healing attitude plays a vital role in performing effective customer relationship management with both parties.

7. CONCLUSIONS AND SCOPE FOR FUTURE WORK

In this paper, we have performed a comprehensive survey about various research works pertain to service discovery, service composition and provisioning of SLA from both SOA and Cloud perspective. Based on the literature survey, emerging research issues have been tailored for further exploration towards achieving service excellence in cloud computing paradigm. Besides, we have highlighted the need for service-oriented broker in cloud computing for augmenting cloud service brokerage with all its functional modules. Thus, this paper provides a detailed insight into the services computing paradigm by opening up new research avenues.

Some of the promising research directions in the phase of service discovery and selection are (i) Inclusion of fuzzy logic principles for considering the valid requirements of the user and elimination of in-correct services (ii) Applying machine learning techniques such as decision trees, Artificial Neural Networks for improving the dynamic discovery and selection of appropriate cloud services based on the user's need (iii) Adaptation of Multi criteria decision making methods to assists the user in selecting the services based on feature such as contextual information.

In the phase of service composition, possible research avenues are (i) Interaction of multiple agents by using 'single sign on' identification towards offering secured service, (ii) Elevate the cloud broker with the interoperable nature of SOA for easy interchanging of services without compromising service quality, (iii) Adaptation of producer-consumer service log into the cloud broker for maintaining the trust details of business process.

With respect to the service provisioning phase, the future research directions are (i) Incorporation of MCDM approaches with Matrix Factorization techniques to predict the services with better QoS values and try to resolve the problem of cold-start, (ii) Applying the data analytics tools such as MapReduce and Spark for the processing of user feedbacks to predict the service quality and ensures the utilization of services at the customer end, (iii) Adapting a supervised learning based crawler to notify the cloud user about the service updates, (iv) Designing various efficient meta information model using feature selection methods for efficient SLA documentation for both SOA and cloud.

REFERENCES

[1] L. Liu, "Services Computing: From Cloud Services, Mobile Services to Internet of Services", IEEE Transactions on Services Computing, 9(5), 2016, pp. 661-663.

[2] T. Erl, P. Merson, R. Stoffers, "Service-oriented Architecture: Analysis and Design for Services and Microservices", Prentice Hall PTR, 2017.

[3] T. Erl, "Service-oriented architecture: concepts, technology, and design", Pearson Education India, 2005.

[4] T. Erl, R. Puttini and Z. Mahmood, "Cloud computing: concepts, technology & architecture", Pearson Education, 2013.

[5] M. Manoharan and S. Selvarajan, "An efficient methodology to improve service negotiation in cloud environment", International Journal of Grid and Utility Computing, 6(3-4), 2015, pp. 150-158.

[6] S.F. Chang, "A reference architecture for application marketplace service based on SaaS", International Journal of Grid and Utility Computing, 2(4), 2011, pp. 243-252.

[7] Y. Wei and M.B. Blake, "Service-oriented computing and cloud computing: Challenges and opportunities", IEEE Internet Computing, 14(6), 2010, pp. 72-75.

[8] M. Aazam and E.N. Huh, "Cloud broker service-oriented resource management model", Transactions on Emerging Telecommunications Technologies, 28(2), 2017.

[9] K. Sycara, M. Paolucci, J. Soudry and N. Srinivasan, "Dynamic discovery and coordination of agent-based semantic web services", IEEE Internet computing, 8(3), 2004, pp. 66-73.

[10] L. Pan, "Towards a ramework for automated service negotiation in cloud computing", In: IEEE International Conference on Cloud Computing and Intelligence Systems (CCIS), 2011, pp. 364-367.

[11] H. Dong, F.K. Hussain and E. Chang, "A service concept recommendation system for enhancing the dependability of semantic service matchmakers in the service ecosystem environment", Journal of Network and Computer Applications, 34(2), 2011, pp. 619-631.

[12] M. Crasso, A. Zunino and M. Campo, "Easy web service discovery: A query-by-example approach", Science of Computer Programming, 71(2), 2008, pp. 144-164.

[13] T. Rajendran, P. Balasubramanie and R. Cherian, "An optimal broker-based architecture for web service discovery with QoS

characteristics", International Journal of Web Services Practices, 5(1), 2010, pp. 32-40.

[14] R.R. Yager, G. Gumrah and M.Z. Reformat, "Using a web Personal Evaluation Tool–PET for lexicographic multi-criteria service selection", Knowledge-Based Systems, 24(7), 2011, pp. 929-942.

[15] A.P. De Souza and R.J. Rabelo, "A model for dynamic services discovery over largely distributed providers based on QoS and business processes contexts", In: IEEE World Congress on Services, 2011, pp. 347-354.

[16] Z. Khan, M. Odeh and R. McClatchey, "Bridging the gap between business process models and service-oriented architectures with reference to the grid environment", International Journal of Grid and Utility Computing, 2(4), 2011, pp. 253-283.

[17] J.M. Garcia, D. Ruiz and A. Ruiz-Cortes, "Improving semantic web services discovery using SPARQL-based repository filtering", Web Semantics: Science, Services and Agents on the World Wide Web, 17, 2012, pp. 12-24.

[18] J. Zhang, J. Wang, P.C. Hung, Z. Li, J. Liu and K. He, "Leveraging Fragmental Semantic Data to Enhance Services Discovery", Information System Frontiers (ISF), 2012.

[19] S.K. Mistry, M.H. Kamal and D. Mistry, "Semantic discovery of web services through social learning", Procedia Technology, 3, 2012, pp. 167-177.

[20] Y. Peng, "Rapid Service Discovery Based on Clustering", Physics Procedia, 33, 2012, pp. 195-200.

[21] C.R. Choi and H.Y. Jeong, "A broker-based quality evaluation system for service selection according to the QoS preferences of users", Information Sciences, 277, 2014, pp. 553-566.

[22] R. Ramacher and L. Monch, "Service selection with runtime aspects: A hierarchical approach", IEEE Transactions on Services Computing, 8(3), 2015, pp. 481-493.

[23] N. Zhang, J. Wang, Y. Ma, K. He, Z. Li and X.F. Liu, "Web service discovery based on goal-oriented query expansion", Journal of Systems and Software, 142, 2018, pp. 73-91.

[24] M. Parhi, B. K. Pattanayak and M. R. Patra, "A multi-agent-based framework for cloud service discovery and selection using ontology", Service Oriented Computing and Applications, 12(2), 2018, pp. 137-154.

[25] M. Parhi, B. K. Pattanayak and M. R. Patra, "An ontology-based cloud infrastructure service discovery and selection system", International Journal of Grid and Utility Computing, 9(2), 2018, pp. 108-119.

[26] H. Nabli, R. B. Djemaa and I. A. B. Amor, "Efficient cloud service discovery approach based on LDA topic modeling", Journal of Systems and Software, 146, 2018, pp. 233-248.

[27] R. Nagarajan, R. Thirunavukarasu, S. Shanmugam, "A cloud broker framework for infrastructure service discovery using semantic network", International Journal of Intelligent Engineering and Systems, 11(3), 2018, pp. 11-19.

[28] F. Nawaz, M.R. Asadabadi, N. K. Janjua, O. K. Hussain, E. Chang and M. Saberi, "An MCDM method for cloud service selection using a Markov chain and the best-worst method", Knowledge-Based Systems, 159, 2018, pp. 120-131.

[29] R.R. Kumar, S. Mishra and C. Kumar, "A novel framework for cloud service evaluation and selection using hybrid MCDM methods", Arabian Journal for Science and Engineering, 43(12), 2018, pp. 7015-7030.

[30] L. Sun, H. Dong, O. K. Hussain, F. K. Hussain and A. X. Liu, "A framework of cloud service selection with criteria interactions", Future Generation Computer Systems, 94, 2019, pp. 749-764.

[31] A. Al-Faifi, B. Song, M. M, Hassan, A. Alamri and A. Gumaei, "A hybrid multi criteria decision method for cloud service selection from Smart data. Future Generation Computer Systems, 93, 2019, pp. 43-57.

[32] G. Nie, Q. She and D. Chen, "Evaluation index system of cloud service and the purchase decision-making process based on AHP", In: Proceedings of the 2011 International Conference on Informatics, Cybernetics, and Computer Engineering (ICCE2011), Melbourne, Australia, 2011, pp. 345-352.

[33] S. Ding, Z. Wang, D. Wu and D.L. Olson, "Utilizing customer satisfaction in ranking prediction for personalized cloud service selection", Decision Support Systems, 93, 2017, pp. 1-10.

[34] H. Mezni and T. Abdeljaoued, "A cloud services recommendation system based on Fuzzy Formal Concept Analysis", Data & Knowledge Engineering, 116, 2018, pp. 100-123.

[35] R. Nagarajan, S. Selvamuthukumaran, R. Thirunavukarasu, "A fuzzy logic based trust evaluation model for the selection of cloud services", In: IEEE Proc. of international conference on computer communication and informatics, 2017.

[36] R. Nagarajan, R. Thirunavukarasu, S. Shanmugam, "A fuzzy-based intelligent cloud broker with mapreduce framework to evaluate the trust level of cloud services using customer feedback", International Journal of Fuzzy Systems, 20(1), 2018, pp. 339-347.

[37] R. Nagarajan, R. Thirunavukarasu, "A fuzzy-based decision-making broker for effective identification and selection of cloud infrastructure services", Soft Computing, 23(19), 2019, pp. 9669-9683.

[38] K. J. Modi and S. Garg, "A QoS-based approach for cloud-service matchmaking, selection and composition using the Semantic Web", Journal of Systems and Information Technology, 21(1), 2019, pp. 63-89.

[39] S. Pang, Q. Gao, T. Liu, H. He, G. Xu and K. Liang, "A Behavior Based Trustworthy Service Composition Discovery Approach in Cloud Environment", *IEEE Access*, 7, 2019, pp. 56492-56503.

[40] H. Ma, Z. Hu, K. Li and H. Zhu, "Variation-aware Cloud Service Selection via Collaborative QoS Prediction. IEEE Transactions on Services Computing, 2019.

[41] Y. Wang, J. Wen, W. Zhou, B. Tao, Q. Wu and Z. Tao, "A Cloud Service Selection Method Based on Trust and User Preference Clustering", IEEE Access, 7, 2019, pp. 110279-110292.

[42] H. Qian, H. Zu, C. Cao and Q. Wang, "CSS: Facilitate the cloud service selection in IaaS platforms", In: International Conference on Collaboration Technologies and Systems (CTS), 2013, pp 347-354.

[43] C. Mastroianni, G. Papuzzo, "A self-organizing P2P framework for collective service discovery", Journal of Network and Computer Applications, 2014, 39, 214-222.

[44] A. Zhygmanovskyi and N. Yoshida, "Cloud service provisioning based on peer-to-peer network for flexible service sharing and discovery", Journal of Computer and Communications, 2(10), 2014, pp. 17.

[45] M. Tang, X. Dai, J. Liu and J. Chen, "Towards a trust evaluation middleware for cloud service selection", Future Generation Computer Systems, 74, 2017, pp. 302-312.

[46] H.Y. Paik, A.L. Lemos, M.C. Barukh, B. Benatallah and A. Natarajan, "Web Service Composition: Overview", In: Web Service Implementation and Composition Techniques, 2017, pp. 149-158.

[47] W. Niu, G. Li, H. Tang, X. Zhou and Z. Shi, "CARSA: A context-aware reasoning-based service agent model for AI planning of web service composition", Journal of Network and Computer Applications, 34(5), 2011, pp. 1757-1770.

[48] M.B. Blake, "Coordinating multiple agents for workflow-oriented process orchestration", Information Systems and E-Business Management, 1(4), 2003, pp. 387-404.

[49] S. Wang, W. Shen and Q. Hao, "An agent-based Web service workflow model for inter-enterprise collaboration", Expert Systems with Applications, 31(4), 2006, pp. 787-799.

[50] A.E. Walsh, “UDDI, SOAP, and WSDL: the web services specification reference book”, Prentice Hall Professional Technical Reference, 2002.

[51] V. Korupala, A. Kothalanka and S. Gandi, “A Novel Architecture for Dynamic Invocation of Web Services”, In: Proceedings of the 48th Annual Convention of Computer Society of India-Vol II, Springer International Publishing, 2014, pp. 671-678.

[52] W. Chen and I. Paik, “Toward better quality of service composition based on a global social service network”, IEEE Transactions on Parallel and Distributed Systems, 26(5), 2015, pp. 1466-1476.

[53] N.H. Rostami, E. Kheirkhah and M. Jalali, “An optimized semantic web service composition method based on clustering and ant colony algorithm”, International Journal of Web & Semantic Technology, 2014.

[54] Q.Z. Sheng, X. Qiao, A.V. Vasilakos, C. Szabo, S. Bourne and X. Xu, “Web services composition: A decade’s overview”, Information Sciences, 280, 2014, pp. 218-238.

[55] Z. Li, L. O’Brien and H. Zhang, “Circumstantial-evidence-based effort judgement for web service composition-based SOA implementations”, International Journal of Space-Based and Situated Computing, 2(1), 2012, pp. 31-44.

[56] M. Ghobaei-Arani, A. A. Rahmanian, M. S. Aslanpour and S. E. Dashti, “CSA-WSC: cuckoo search algorithm for web service composition in cloud environments”, Soft Computing, 22(24), 2017, pp. 8353-8378.

[57] G. Zou, Y. Chen, Y. Yang, R. Huang and Y. Xu, “AI planning and combinatorial optimization for web service composition in cloud computing”, In: Proceeding of International conference on cloud computing and virtualization, 2010, pp 1-8.

[58] S. K. Gavvala, C. Jatoth, G. R. Gangadharan and R. Buyya, “QoS-aware cloud service composition using eagle strategy”, Future Generation Computer Systems, 90, 2019, pp. 273-290.

[59] A. Naseri and N. J. Navimipour, “A new agent-based method for QoS-aware cloud service composition using particle swarm optimization algorithm”, Journal of Ambient Intelligence and Humanized Computing, 10(5), 2019, pp. 1851-1864.

[60] A. Merizig, O. Kazar and M. Lopez-Sanchez, “A dynamic and adaptable service composition architecture in the cloud based on a multi-agent system”, International Journal of Information Technology and Web Engineering (Engineering), 13(1), 2018, pp. 50-68.

[61] J.O. Garcia and K.M. Sim, “Agent-based cloud service composition”, Applied Intelligence, 38(3), 2013, pp. 436-464.

[62] K.R. Lee, M.H. Fu and Y.H. Kuo, “A hierarchical scheduling strategy for the composition services architecture based on cloud computing”, In: 2nd International Conference on Next Generation Information Technology (ICNIT), 2011, pp. 163-169.

[63] Y. Wei and M.B. Blake, “Adaptive Web Services Monitoring in Cloud Environments”, International Journal of Web Portals, 5(1), 2013, pp. 15-27.

[64] I. Breskovic, C. Haas, S. Caton and I. Brandic, “Towards self-awareness in cloud markets: A monitoring methodology”, In: IEEE Ninth International Conference on Dependable, Autonomic and Secure Computing (DASC), 2011, pp. 81-88.

[65] P. Patel, A.H. Ranabahu and A.P. Sheth, Service level agreement in cloud computing, 2009.

[66] S.A. Baset, “Cloud Service Level Agreement”, Encyclopedia of Cloud Computing, 2016, pp. 433.

[67] C.A.B. De Carvalho, R.M. De Castro Andrade, M.F. De Castro, E.F. Coutinho and N. Agoulmine, “State of the art and challenges of security SLA for cloud computing”, Computers & Electrical Engineering, 2017.

[68] S. Son and S.C. Jun, “Negotiation-based flexible SLA establishment with SLA-driven resource allocation in cloud computing”, In: 13th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid), 2013, pp. 168-171.

[69] N. Ghosh and S.K. Ghosh, “An approach to identify and monitor SLA parameters for storage-as-a-service cloud delivery model”, In: IEEE Globecom Workshops (GC Wkshps), 2012, pp. 724-729.

[70] G. Wei, A.V. Vasilakos, Y. Zheng and N. Xiong, “A game-theoretic method of fair resource allocation for cloud computing services”, The Journal of Supercomputing, 54(2), 2010, pp. 252-269.

[71] M. Alhamad, T. Dillon and E. Chang, “Conceptual SLA framework for cloud computing”, In: 4th IEEE International Conference on Digital Ecosystems and Technologies (DEST), 2010, pp. 606-610.

[72] M. Alhamad, T. Dillon and E. Chang, “SLA-based trust model for cloud computing”, In: 13th International Conference on Network-Based Information Systems (NBiS), 2010, pp. 321-324.

[73] O. Niehorster, A. Brinkmann, G. Fels, J. Kruger and J. Simon, “Enforcing SLAs in scientific clouds”, In: IEEE International Conference on Cluster Computing (CLUSTER), 2010, pp. 178-187.

[74] R. Buyya, S.K. Garg and R.N. Calheiros, “SLA-oriented resource provisioning for cloud computing: Challenges, architecture, and solutions”, In: International Conference on Cloud and Service Computing (CSC), 2011, pp. 1-10.

[75] P. Wieder, J.M. Butler, W. Theilmann and R. Yahyapour, “Service level agreements for cloud computing”, Springer Science & Business Media, 2011.

[76] M. Torkashvan and H. Haghighi, “CSLAM: A framework for cloud service level agreement management based on WSLA”, In: Sixth International Symposium on Telecommunications (IST), 2012, pp. 577-585.

[77] N.S. Rao, S.W. Poole, F. He, J. Zhuang, C.Y. Ma and D.K. Yau, “Cloud computing infrastructure robustness: A game theory approach”, In: International Conference on Computing, Networking and Communications (ICNC), 2012, pp. 34-38.

[78] A. Al Falasi, M.A. Serhani and R. Dssouli, “A model for multi-levels SLA monitoring in federated cloud environment”, In: IEEE 10th International Conference on Ubiquitous Intelligence and Computing, and 10th International Conference on Autonomic and Trusted Computing (UIC/ATC), 2013, pp. 363-370.

[79] G. Nie, E. Xueni and D. Chen, “Research on service level agreement in cloud computing”, In: Advances in Electric and Electronics, Springer Berlin Heidelberg, 2012, pp. 39-43.

[80] A. Keller and H. Ludwig, “The WSLA framework: Specifying and monitoring service level agreements for web services”, Journal of Network and Systems Management, 11(1), 2003, pp. 57-81.

[81] L. Wu, S.K. Garg, S. Versteeg and R. Buyya, “SLA-based resource provisioning for hosted software-as-a-service applications in cloud computing environments”, IEEE Transactions on services computing, 7, 2014, pp. 465-485.

[82] M. Anisetti, C.A. Ardagna, P.A. Bonatti, E. Damiani, M. Faella, C. Galdi and L. Sauro, “e-Auctions for multi-cloud service provisioning”, In: IEEE International Conference on Services Computing (SCC), 2014, pp. 35-42.

[83] H. Chen, X. Liu, H. Xu and C. Wang, “A Cloud Service Broker Based on Dynamic Game Theory for Bilateral SLA Negotiation in Cloud Environment”, International Journal of Grid and Distributed Computing, 9(9), 2016, pp. 251-268.

[84] F. Messina, G. Pappalardo, C. Santoro, D. Rosaci and G.M. Sarne, “A multi-agent protocol for service level agreement negotiation in cloud federations”, International Journal of Grid and Utility Computing, 7(2), 2016, pp. 101-112.

[85] R. Nagarajan and R. Thirunavukarasu, “A Service Context-Aware QoS Prediction and Recommendation of Cloud Infrastructure Services”, Arabian Journal for Science and Engineering, Springer, 45(4), 2020, pp. 2929-2943.

[86] R. Nagarajan and R. Thirunavukarasu, "A Review on Intelligent Cloud Broker for Effective Service Provisioning in Cloud", In 2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS), 2018, pp. 519-524.

[87] N. Rajganesh and T. Ramkumar, "A review on broker based cloud service model", Journal of computing and information technology, 24(3), 2016, pp. 283-292.

Rajganesh Nagarajan is currently working as an Assistant Professor [SL] in the Department of Information Technology, A.V.C. College of Engineering, Tamilnadu, India. He has completed his Ph.D with Anna University, Chennai. His research interests are Services computing, Machine Learning and Big Data Analytics.

Ramkumar Thirunavukarasu is working as an Associate Professor & HOD in School of Information Technology & Engineering, Vellore Institute of Technology, India. He has received his Ph. D from Anna University, Chennai. His area of specialization includes knowledge discovery from multiple databases and big data analytics with machine learning.