Computational Intelligence in Cloud Computing

Ruggero Donida Labati, Angelo Genovese, Vincenzo Piuri, Fabio Scotti, and Sarvesh Vishwakarma

Abstract Cloud Computing (CC) is a model that enables ubiquitous, convenient, and on-demand network access to a shared pool of configurable computing resources. In CC applications, it is possible to access both software and hardware architectures remotely and with little or no knowledge about their physical or logical locations. Due to its low deployment and management costs, the CC paradigm is being increasingly used in a wide variety of online services and applications, including remote computation, software-as-a-service, off-site storage, entertainment, and communication platforms. However, several aspects of CC applications, such as system design, optimization, and security issues, have become too complex to be efficiently treated using traditional algorithmic approaches under the increasingly high complexity and performance demands of current applications. Recently, advances in Computational Intelligence (CI) techniques have fostered the development of intelligent solutions for CC applications. CI methods such as artificial neural networks, deep learning, fuzzy logic, and evolutionary algorithms have enabled improving CC paradigms through their capabilities of extracting knowledge from high quantities of real-world data, thus further optimizing their design, performance, and security with respect to traditional techniques. This chapter introduces recent CI techniques, reviews the main applications of CI in CC, and presents challenges and research trends.

1 Introduction

Cloud Computing (CC) is an architecture model and computing paradigm that uses parallel and distributed systems composed of several physical and virtual machines to provide a unified shared computing resource in a runtime environment by aggregating resources and offering remote users a single system view. By using a CC

Department of Computer Science, Università degli Studi di Milano
via Celoria 18, I-20133 Milan (MI), Italy.
e-mail: *firstname.lastname*@unimi.it

infrastructure, hardware and software services can be delivered to remote users in a ubiquitous and on-demand manner. The main advantage of CC resides in allowing anyone connected to the internet to use the shared computing resources and services according to their needs through a convenient pay-per-use basis, irrespective of where the services are hosted [9]. In fact, the main vendors that provide CC services give users the illusion of infinite computing resources and the ability to pay for their use as needed [80].

Thanks to these features, the use of CC is currently experiencing a growth rate of 50% per year in delivering scalable and on-demand services over the internet to an increasing number of users [91]. At present, there are approximately 75 million servers that are active worldwide [77], with cloud-based servers representing a significant portion of the total number of servers [33]. It has recently been estimated that 70% of companies are using CC services [46], and it has been predicted that by the year 2020, there may be a need to include 400 million additional servers in data centers to facilitate the increasing need for computing services [20].

To efficiently provide services as needed, cloud vendors need to optimize the available resources, both during the system design and at runtime, by understanding the resource availability in complex situations, such as situations involving several end users of CC services whose future needs may be unknown. Therefore, vendors have the need to predict the use of the computing architecture, storage space, and network bandwidth to enable a dynamic scalability of the utilization by adapting workload distribution, disk provisioning, and network capacity [15].

Currently, with the increasing complexity of CC architectures, the high performance demands of current applications, and the increasing number of end users with no up-front commitment, it is often difficult to accurately predict the CC resources requested by end users at runtime and thus to dynamically provide the services as needed [99]. In some cases, system design and resource optimization in new-generation CC environments have become very difficult to theoretically model using traditional algorithmic methods, exposing them to inefficient workload distributions, wasteful power management, or security issues [18].

To accurately predict and optimize cloud resource utilization in complex and dynamic environments, recent methods are considering intelligent solutions based on Computational Intelligence (CI) techniques, such as Artificial Neural Networks (ANNs), Support Vector Machines (SVMs), fuzzy logic, Evolutionary Algorithms (EAs), and Deep Learning (DL). In fact, CI techniques have the advantages of being able to process large quantities of data to learn complex relations by examples; thus, these techniques do not require theoretical modeling, can adapt to variations in operating parameters, and are robust to noise. Thanks to these characteristics, CI-based methods have been successfully applied for event prediction [26,54], modeling of complex phenomena [24], and optimization [8]. In CC applications, CI techniques are used for heterogeneous problems, including resource optimization [18, 62] and protection of privacy and security [50, 65, 79].

To highlight the recent advances in designing and implementing new-generation CC services based on CI techniques, this paper provides an overview of the CI techniques that are used for CC applications. The remainder of this paper is organized as follows. Section 2 presents an overview of the main CI techniques proposed in the literature. Section 3 introduces the main cloud services that are commercially available. Section 4 describes the applications of CI in the design and optimization of CC systems. Section 5 discusses the challenges and future research trends. Finally, Section 6 concludes this work.

2 Overview of Computational Intelligence Techniques

CI techniques enable an intelligent behavior in information processing architectures by learning complex phenomena from examples, adapting to variations in operating conditions, and being robust to noisy and incomplete information. The main advantage of CI methods is the ability to process large quantities of data to provide approximate, robust, and adaptive solutions, in most cases with a lower computational complexity than using exact models [27]. These aspects are contributing to the increasing use of CI models in several application scenarios [24, 25, 48], including CC applications [18, 62].

This section presents an overview of the CI techniques that are mostly used in designing and optimizing CC systems. Specifically, this section reviews the main aspects of ANNs, DL, SVMs, fuzzy logic, and EAs [11, 48]:

- *ANNs* consist of layers of artificial neurons that process information using a transfer function and that are interconnected based on a specific network topology, with each connection associated with a weight and a bias [3]. During the training step, ANNs use a learning procedure to adapt weights and biases such that the network can model an approximate relation between the input samples and the corresponding outputs. By using a high number of training samples, ANNs can learn a model that generalizes the observed phenomenon, is able to process new inputs, and provides approximate solutions with robustness to noise and low computational complexity. Several types of ANNs have been proposed in the literature, which are classified based on the transfer function used and the topology of the network, such as feedforward neural networks, self-organizing maps, and radial basis function networks [31, 39].
- *DL* is a recent evolution of ANNs obtained by using a larger number of hidden layers of neurons and dedicated training algorithms. DL techniques frequently achieve better accuracy than ANNs and SVMs due to their capability of learning data representations without a handcrafted feature extraction step. However, DL techniques generally require longer training times than ANNs, as well as vast training datasets to be properly tuned. Examples of widely used DL techniques are convolutional neural networks, deep belief networks and stacked autoencoders [11, 59].
- *SVMs* are kernel-based classifiers that project the input data onto a higher-dimensionality space, with the purpose of increasing the separability of the classes and therefore facilitating the learning procedure. SVMs have the main advantage of using a learning function based on the optimization of a convex

surface, which avoids incurring local optima. In addition, SVMs have a limited number of parameters and therefore have a more straightforward tuning process than ANNs [14].

- *Fuzzy logic* represents an intelligent computing approach that can process data when also expressed with different degrees of uncertainty, similar to what occurs with human reasoning processes. Due to this peculiarity, fuzzy logic is particularly used in cases of imprecise or incomplete data, for example, when dealing with language processing, optimizing conflicting objectives, or designing knowledge bases [93].
- *EAs* are optimization methods that compute a global solution by imitating biological evolution. Methods based on EA start by computing an initial population of solutions (e.g., using a random initialization or heuristics tailored to the application scenario) that evolve and can interact between themselves, iteratively converging toward a better solution. To improve the quality of the solutions at each step, EAs use methods inspired by biological processes and natural evolution, such as crossover, mutation, or selections. Based on the level of interactions and the operators used, different EAs have been proposed in the literature, including Genetic Algorithms (GA), swarm intelligence, and differential evolution [88].

3 Commercial Cloud Computing Services

CC vendors are currently offering a wide spectrum of cloud-based services that include infrastructure resources, programming platforms, and common software functionalities, with the purpose of enabling the remote utilization and management of several computing aspects, such as data processing, storage, databases, networking, data management, business analytics, developing tools, and security. In particular, CC services are currently organized and delivered to end users according to three main paradigms:

- *Infrastructure as a Service (IaaS)* has the purpose of arranging, emulating, and offering remote users on-demand infrastructure resources, such as servers, storage, and networking.
- *Platform as a Service (PaaS)* has the purpose of offering a programming platform and a runtime environment for developing, debugging, testing, and maintaining applications.
- *Software as a Service (SaaS)* has the purpose of providing common software functionalities to users, such as document management, photo editing, office automation, email utilities, social networking, and customer relationship management, through a web service running on a CC infrastructure.

In many cases, the combined utilization of IaaS, PaaS, and SaaS allows end users and companies to fully outsource their data processing centers to the cloud, particularly when security aspects are handled by the CC provider [80].

Table 1 Overview of the cloud services offered by main commercial cloud vendors

Ref.	Cloud service	Paradigm	Deployment model	Challenges
[43]	IBM Cloud	IaaS PaaS SaaS	Private Public Hybrid	Power consumption [42] Workload optimization [45] Corporate security [44]
[35]	Google Cloud Platform	IaaS PaaS SaaS	Private Public Hybrid	Power consumption [21] Workload optimization [81]
[4]	Amazon Web Services	IaaS	Private Public Hybrid	File security [32] Storage optimization [6] User cost optimization [5]
[69]	Microsoft Azure	PaaS	Private Public Hybrid	File security [68] Workload optimization [67]
[74]	Oracle Cloud	IaaS PaaS SaaS	Private Public Hybrid	Workload optimization [73] Corporate security [90]

Notes. IaaS: Infrastructure as a Service; PaaS: Platform as a Service; SaaS: Software as a Service.

Based on the locations of the CC data center, cloud services can also be classified into three deployment models: *i)* private cloud, in which the cloud is implemented within the private company; *ii)* public cloud, when the cloud is available to the general public; and *iii)* hybrid cloud, representing a combination of private and public clouds [9].

Several major companies are currently offering on a large scale private, public, or hybrid CC services based on IaaS, PaaS, or SaaS. Table 1 describes the main commercial services offered by the top seven CC vendors, along with their paradigms, deployment models, and challenges. The vendors are selected based on the capacity of their data centers, popularity of the offered services, and annual revenue in market share [97]. In particular, the main commercial cloud services include IBM Cloud [43], Google Cloud Platform (GCP) [35], Amazon Web Services (AWS) [4], Microsoft Azure [69], and Oracle Cloud [74].

As shown in Table 1, CC vendors are currently experiencing several challenges in offering efficient services to end users. In particular, the problem of predicting and then optimizing the workload is one the main challenges and is faced by most vendors for delivering scalable performance [56]. Meanwhile, several vendors are considering challenges related to high power consumption [61] and guaranteeing the security of data, especially when dealing with corporate information [49]. Several vendors are therefore increasingly considering the use of CI techniques toward reducing these problems, for example, by using DL techniques to optimize the use of resources [21] and reduce costs [22].

Table 2 Examples of Computational Intelligence Techniques used in Cloud Computing systems

Application	Problem	CI techniques				
		ANN	**DL**	**SVM**	**FL**	**EA**
Resource optimization	Cloud brokering	-	-	-	-	[47, 51, 57]
	VM placement in private clouds	-	-	-	-	[86, 95]
	Service composition and placement	-	-	-	-	[30, 63]
	Workflow scheduling	-	-	-	-	[17, 36, 38]
	Capacity planning	[72]	[71]	-	-	[34]
	Server farm load balancing	-	[52, 64]		[87]	[58]
Supporting security and privacy	Security	[40, 50] [53, 55]	[89, 92]	-	-	-
	Privacy	-	[65]	[79]	-	-

Notes. ANN: Artificial Neural Network; DL: Deep Learning; SVM: Support Vector Machine; FL: Fuzzy Logic; EA: Evolutionary Algorithm; VM: Virtual Machine.

4 Computational Intelligence Techniques in Cloud Computing

Complex and dynamic infrastructures such as CC systems can generate vast amounts of data, which frequently require the use of CI techniques to be processed with satisfactory accuracy and low computational time. CI techniques are being increasingly used in CC applications since they can model complex phenomena by learning from examples while simultaneously processing vast amounts of noisy data with relatively fast computations [76]. CI techniques are therefore achieving increasing importance in designing, maintaining, and optimizing CC systems. In particular, CI techniques are widely used to optimize resources and to realize privacy and security protection mechanisms. Furthermore, CI techniques can be used in CC applications to infer additional information from databases [70] and to analyze network traffic [29].

In this section, we briefly overview the main applications of CI techniques for designing, maintaining, and optimizing CC systems. Table 2 provides some significative examples of CI techniques that have been proposed in the literature for different applications and problems of CC systems. In particular, we consider CI-based techniques applied for resource optimization and used for supporting privacy and security.

4.1 Resource Optimization

In CC applications, distributed hardware and software resources are aggregated and shared to offer end users a single system view that is accessible on-demand [9]. Examples of shared resources include computing servers, operating systems, networks, software applications, and storage equipment.

The CC resource optimization problems can be classified as static (or offline) and dynamic (or online) problems according to the considered time frame. In the first case, the complete list of requests is known in advance, while in the second case, the resource requests can initially be unknown and change over time. Since CC applications are typically on-demand services, most of the optimization problems are dynamic, with some of the resource optimization aspects, such as performing batch scheduling, based on static approaches.

Due to the necessity of allocating resources for a potentially long time span and on a distributed architecture, most of the CC resource optimization problems are NP-hard [86]. In CC applications, to guarantee fast and reliable solutions even when dealing with NP-hard problems in both static and dynamic cases, many studies in the literature consider using CI techniques. In particular, EAs are widely used to solve combinatorial problems with satisfactory results. However, these techniques can require longer computational times than trained ANNs, DL, or fuzzy systems. Therefore, ANNs, DL, fuzzy logic, and hybrid approaches are currently more suitable for solving dynamic optimization problems [37].

In the following, we describe how CI techniques are applied to solve both static and dynamic resource optimization problems.

4.1.1 Static Problems

The main static resource optimization methods for CC applications include cloud brokering, Virtual Machine (VM) placement, service composition, and workflow scheduling.

- *Cloud brokering* consists of matching multiple requests from multiple users to the offers of multiple cloud vendors. This process can be considered as an optimization problem with the main goal of minimizing the costs, often coupled with the secondary goal of maximizing the quality of service (QoS). In particular, a cloud broker is an intermediary in managing the CC infrastructure, which can help in providing access to a wide set of services and in optimizing the resource selection. In the literature, there are two main classes of methods based on CI for cloud brokering. The first class is based on the assumption that cloud vendors are offering a homogeneous infrastructure and that the pool of resources is unlimited. This type of method is used to select a type of VM for each computational task. Methods based on GAs are primarily used to handle this problem [51]. The second class of methods assumes a fixed set of resources or reserved instances of the broker, with fixed numbers and types of machines. EAs also represent the most used techniques in this class of method. Examples of treated problems include maximizing the revenue of the cloud broker using a hybrid EA [47] and performing a multiobjective optimization for connected Internet of Things (IoT) using particle swarm optimization [57].

- *Virtual machine placement in private clouds* consists of choosing the best placement of VMs for previously defined hardware infrastructures. Similar to cloud brokering, most of the approaches in the literature use a solution vector computed by mapping VMs to physical machines. In fact, to efficiently aggregate, allocate, and relocate a set of hardware and software resources, CC environments often use VMs, represented by a set of logical resources that can be mapped to different physical hardware. In the literature, there are studies of VM placement based on different EAs, for example, based on ant colony optimization [86], and on GAs with a multiobjective fuzzy evaluation [95].
- *Service composition and placement* aims to define the best allocation of a complex service onto a set of available virtualized resources. The considered resources can be heterogeneous and can include hardware components, VMs, and software modules. The optimization problem does not require a real-time computation and can therefore be performed offline. Methods in the literature typically aim to optimize both the costs and the QoS, primarily using approaches based on GAs [30, 63].
- *Workflow scheduling* aims to optimally map the workflow tasks to VMs according to functional and nonfunctional requirements. A workflow consists of a series of interdependent tasks that are bound together through data or functional dependencies, which should be respected in the scheduling process. Some common objectives of the workflow scheduling process consider optimizing the load balancing, minimizing the costs and the makespan, respecting the deadlines and service-level agreements, and guaranteeing the availability, reliability, and security. In the literature, there are many studies on this topic based on different approaches [66], which can be based on heuristics, CI techniques, or hybrid approaches. The most used CI techniques are based on EAs, such as GAs [38], ant colony optimization [17], and swarm optimization [36].

4.1.2 Dynamic Problems

The main dynamic resource optimization methods for CC applications include capacity planning and server farm load balancing.

- *Capacity planning* consists of predicting the future load of a CC system and should be performed dynamically to properly match the user requests with the existing resources. Since waiting times for obtaining resources can affect the user satisfaction and QoS, while excessive resource allocation can increase energy consumption, the main goal of capacity planning approaches is to optimize the QoS while minimizing the costs. In the literature, there are different methods for capacity planning based on CI. In particular, EAs can be used for predicting the future load of a CC system [34], but they can require long computational times. In contrast, CI-based approaches based on ANNs [72] or DL [71] feature a reduced computational time compared to methods based on EAs and can therefore be executed with higher frequency.

- *Server farm load balancing* is responsible for optimizing the dispatch of incoming requests from the users to the available machines. It is generally performed online and is particularly relevant for SaaS providers. The main goal is to guarantee the expected QoS while minimizing the use of resources, costs, and energy. In the literature, the methods for load balancing the server farm can be divided into centralized or distributed methods [96] and include techniques based on EAs [58], neuro-fuzzy systems [87], deep belief neural networks [52], and deep reinforcement learning [64].

4.2 Supporting Security and Privacy

Guaranteeing data security means ensuring authentication, data integrity, confidentiality, and nonrepudiation. In addition, ensuring privacy also requires deploying mechanisms for data protection [83]. In CC systems, data security and privacy are particularly important since private data are frequently outsourced to companies that control their storage and own the computers used for processing. For this reason, security and privacy concerns can limit the adoption of CC applications for the core business of companies and for applications using sensible data [80]. Consequently, with the increasing utilization of cloud-based applications, several methods in the literature are dealing with guaranteeing high levels of security and privacy in CC environments [49].

To guarantee security in CC systems, it is possible to increase the trust on authentication mechanisms with respect to traditional systems using password-based schemes by applying biometric identity recognition techniques based on DL [89]. To improve security by ensuring the availability of cloud resources, CI techniques can be used to detect Distributed Denial of Service (DDoS) attacks, which can make the resources unavailable for an indefinite amount of time. As an example, ANNs can be trained to estimate the presence of DDoS from data traffic on the network [50]. CI techniques can also be adopted to detect cloud malware attacks, which are attacks on the cloud control interface, where the hacker may gain control of user's private data stored in the cloud. In particular, methods based on ANNs and DL are used in the literature to recognize cloud malware attacks by analyzing data traffic on the network [40, 53, 55, 92].

To protect the privacy of the data, CI techniques can also be used in conjunction with other data-protection approaches to perform computations in the encrypted domain, thus not revealing information to the server. For example, a protocol for performing classification in the encrypted domain based on SVM and homomorphic cryptosystems can protect classification requests [79], while a protocol based on DL and multiparty computation can also be adopted to preserve data privacy in CC applications [65].

5 Challenges and Research Trends

The increasing uptake of CC technologies and the rise of computationally intensive application scenarios requiring cloud-based services (e.g., DL and big data applications [16]) are constantly posing new challenges related to increasing computation, bandwidth, and storage requirements. Such challenges are requiring vendors to continue updating their hardware and software architectures to not only meet current demands but also be prepared for future needs [94].

Together with the increased use of CC services, emerging communication and processing paradigms, such as the IoT or high-speed mobile connections, are introducing new challenges and research trends related to their integration with CC. For example, the fog/edge computing paradigm is emerging in IoT applications as a way to mediate between remote CC processing and local computation, with the purpose of improving response time [19]. Meanwhile, cloud services that are able to run on low-power architectures are being increasingly researched to take advantage of mobile computing platforms [23].

To face the new challenges caused by the increased use of CC services and the related integration with emerging technologies, the academic and industrial research communities are working toward designing new solutions to improve several aspects of new-generation CC architectures. In particular, current research trends are considering the main following aspects [13]:

- *Scalability* refers to the ability of the cloud vendors to provide services as needed, growing and shrinking the allocated resources to provide the end users the illusion of infinite computing resources.
- *Reliability* is related to the ability of the CC infrastructure to withstand and overcome errors and failures, either hardware (e.g., broken components, network congestion) or software (e.g., system crashes).
- *Sustainability* has the purpose of minimizing the amount of energy consumption of CC servers, both during their manufacturing and during their lifetime.
- *Heterogeneity* considers the interconnection of different resources and technologies at the hardware, software, and vendor levels to provide a unified view of the CC services.
- *Mobile computing* refers to the ability to provide cloud services also on resource-constrained devices, such as smartphone or tablets, while simultaneously taking advantage of their wide range of sensors (e.g., camera, GPS, accelerometers).
- *Security* is related to guaranteeing the confidentiality, integrity, and availability of data and computations processed on the cloud.
- *Usability* considers how easy a CC system is to use and how much time people need to learn it.

Among the aforementioned aspects, CI techniques are being increasingly used to improve the scalability, reliability, and sustainability of current CC architectures, which are highly complex and dynamic [18], by performing an intelligent optimization of the processing resources and the energy consumption. Compared to traditional optimization methods, CI techniques are emerging as a more robust way to predict the utilization of CC services using large quantities of noise-affected data and then allocate the necessary resources in an optimized manner, thereby improving the efficiency in managing cloud services and infrastructures [10, 13, 84]. CI techniques are also being considered to improve security aspects in CC applications [12] by exploiting their ability to detect complex malicious data patterns based on examples [1]. Consequently, some CC vendors are increasingly implementing CI models to increase the level of protection against attacks [28].

In the field of CI techniques, recent advances such as DL are constantly improving the classification accuracy and optimization capabilities of intelligent systems [75]. Recent design methodologies for CC architectures are therefore increasingly considering deep models to optimize the allocation of resources [98], improve energy efficiency [21], and provide an additional layer of security to protect the data of end users [1].

With the increased use of DL models in several aspects of computer science and engineering, CC services are emerging as a cost-effective approach to provide dedicated architectures (e.g., CUDA-enabled GPUs) to develop, test, and execute CI models. CC vendors are therefore offering services based on a combination of IaaS, PaaS, and SaaS to facilitate end users designing and running their own deep models for a wide variety of applications [7, 82].

The use of DL techniques is also emerging in recent computing infrastructures based on the fog/edge paradigm, which extends the CC model by shifting some of the computation from remote servers to local processing architectures [19]. In fog computing infrastructures, recent methods are increasingly considering the use of DL to perform some preliminary processing steps at the local level, with the purpose of reducing network traffic and improving response time [41, 60].

Due to their advantages, the increasing adoption of CI techniques in CC, as well as in related paradigms such as fog/edge computing, is currently fostering the design of intelligent and secure solutions that are able to automatically tune the CC configuration to optimize throughput, workload, and energy costs [2, 78, 85, 98].

6 Conclusions

This chapter provided an overview of Computational Intelligence (CI) technologies in Cloud Computing (CC) systems by describing recent methods, commercial services, applications of CI in CC, challenges, and research trends.

The use of CI technologies for the design and optimization of information processing systems is continuously increasing due to the always higher complexity of network infrastructures, the increasing utilization of remote services, and the recent advances regarding new CI models with improved classification and optimization capabilities. In fact, CI techniques allow learning a model of complex phenomena in an easier way than traditional techniques by providing robust and approximate solutions with low computational time. For these reasons, several vendors are increasingly investing in the use of CI techniques for reducing the complexity of problems due to the fast and constant growth of heterogeneous commercial applications. Among CI techniques, the most widely used include artificial neural networks, support vector machines, deep learning, fuzzy logic, and evolutionary algorithms.

Several studies in the literature consider CI-based methods for designing and managing CC systems, with specific focuses on resource optimization and on security and privacy protection. In the context of resource optimization, the research on CI-based methods involves problems such as cloud brokering, placement of virtual machines in private clouds, service composition and placement, workflow scheduling, capacity planning, and server farm load balancing. In the context of security and privacy protection, the research on CI-based methods involves problems such as authentication techniques, detection of attacks, and computation in the encrypted domain.

The diffusion of recent cloud-based computing paradigms such as fog/edge computing and mobile cloud, along with recent models such as deep learning, is currently motivating the design and implementation of innovative CI-based solutions in a new generation of CC infrastructures and applications.

References

1. Abdelsalam, M., Krishnan, R., Huang, Y., Sandhu, R.: Malware detection in cloud infrastructures using convolutional neural networks. In: Proc. of the 2018 IEEE 11th Int. Conf. on Cloud Computing (CLOUD), pp. 162–169 (2018)
2. Abeshu, A., Chilamkurti, N.: Deep learning: The frontier for distributed attack detection in fog-to-things computing. IEEE Communications Magazine **56**(2), 169–175 (2018)
3. Alippi, C., Ferrero, A., Piuri, V.: Artificial intelligence for instruments and measurement applications. IEEE Instrumentation Measurement Magazine **1**(2), 9–17 (1998)
4. Amazon: Amazon Web Services (AWS) (2018). URL https://aws.amazon.com/whitepapers
5. Amazon: AWS cost optimization (2018). URL https://aws.amazon.com/pricing/cost-optimization/
6. Amazon: AWS storage optimization (2018). URL https://docs.aws.amazon.com/aws-technical-content/latest/cost-optimization-storage-optimization/introduction.html
7. Analytics India Magazine: 10 Machine Learning As A Service (MLaaS) tools for data scientists (2018). URL https://www.analyticsindiamag.com/10-machine-learning-service-mlaas-tools-data-scientists/
8. Ansari, N., Hou, E.: Computational Intelligence for Optimization. Springer Publishing Company, Incorporated (2012)
9. Antonopoulos, N., Gillam, L.: Cloud Computing: Principles, Systems and Applications, 1st edn. Springer Publishing Company, Incorporated (2010)
10. Bankole, A.A., Ajila, S.A.: Predicting cloud resource provisioning using machine learning techniques. In: Proc. of the 2013 26th IEEE Canadian Conf. on Electrical and Computer Engineering (CCECE), pp. 1–4 (2013)
11. Bengio, Y., Courville, A., Vincent, P.: Representation learning: A review and new perspectives. IEEE Trans. on Pattern Analysis and Machine Intelligence **35**(8), 1798–1828 (2013)
12. Bhamare, D., Salman, T., Samaka, M., Erbad, A., Jain, R.: Feasibility of supervised machine learning for cloud security. In: Proc. of the 2016 Int. Conf. on Information Science and Security (ICISS), pp. 1–5 (2016)
13. Buyya, R., Srirama, S.N., Casale, G., Calheiros, R., Simmhan, Y., Varghese, B., Gelenbe, E., Javadi, B., Vaquero, L.M., Netto, M.A.S., Toosi, A.N., Rodriguez, M.A., Llorente, I.M.,

De Capitani di Vimercati, S., Samarati, P., Milojicic, D., Varela, C., Bahsoon, R., Assuncao, M.D.D., Rana, O., Zhou, W., Jin, H., Gentzsch, W., Zomaya, A.Y., Shen, H.: A manifesto for future generation cloud computing: Research directions for the next decade. ACM Computing Surveys **51**(5), 105:1–105:38 (2018)
14. Campbell, C.: An introduction to kernel methods. In: R.J. Howlett, L.C. Jain (eds.) Radial basis function networks: design and applications. Springer Verlag, Berlin (2000)
15. Chaisiri, S., Lee, B., Niyato, D.: Optimization of resource provisioning cost in cloud computing. IEEE Trans. on Services Computing **5**(2), 164–177 (2012)
16. Chen, X., Lin, X.: Big Data Deep Learning: Challenges and perspectives. IEEE Access **2**, 514–525 (2014)
17. Chen, Z., Zhan, Z., Lin, Y., Gong, Y., Gu, T., Zhao, F., Yuan, H., Chen, X., Li, Q., Zhang, J.: Multiobjective cloud workflow scheduling: A multiple populations ant colony system approach. IEEE Trans. on Cybernetics pp. 1–15 (2018)
18. Cheng, H., Yao, X., Yang, S., Zhang, M.: Guest editorial: Special issue on Computational Intelligence for cloud computing. IEEE Trans. on Emerging Topics in Computational Intelligence **2**(1), 1–2 (2018)
19. Chiang, M., Zhang, T.: Fog and IoT: An overview of research opportunities. IEEE Internet of Things Journal **3**(6), 854–864 (2016)
20. Dazeinfo: 400 million new servers might be needed by 2020 (2014). URL https://dazeinfo.com/2014/04/22/internet-comprises-5-million-terabytes-data-weighs-much-grain-sand/
21. DeepMind: DeepMind AI reduces Google data centre cooling bill by 40% (2016). URL https://deepmind.com/blog/deepmind-ai-reduces-google-data-centre-cooling-bill-40/
22. Densify: Cost optimization through machine learning (2018). URL https://www.densify.com/service/cloud-cost-optimization
23. Dinh, H.T., Lee, C., Niyato, D., Wang, P.: A survey of mobile cloud computing: architecture, applications, and approaches. Wireless Communications and Mobile Computing **13**(18), 1587–1611 (2013)
24. Donida Labati, R., Genovese, A., Muñoz, E., Piuri, V., Scotti, F.: Applications of computational intelligence in industrial and environmental scenarios. In: V. Sgurev, V. Piuri, V. Jotsov (eds.) Learning Systems: From Theory to Practice, pp. 29–46. Springer International Publishing, Cham (2018)
25. Donida Labati, R., Genovese, A., Muñoz, E., Piuri, V., Scotti, F., Sforza, G.: Computational intelligence for biometric applications: a survey. International Journal of Computing **15**(1), 40–49 (2016)
26. Donida Labati, R., Genovese, A., Piuri, V., Scotti, F., Sforza, G.: A decision support system for wind power production. IEEE Trans. on Systems, Man, and Cybernetics: Systems pp. 1–15 (2018)
27. Engelbrecht, A.: Computational intelligence: an introduction. Wiley (2007)
28. eWEEK: Google turns to AI, machine learning to improve cloud security (2018). URL http://www.eweek.com/cloud/google-turns-to-ai-machine-learning-to-improve-cloud-security
29. Fadlullah, Z.M., Tang, F., Mao, B., Kato, N., Akashi, O., Inoue, T., Mizutani, K.: State-of-the-art deep learning: Evolving machine intelligence toward tomorrow's intelligent network traffic control systems. IEEE Communications Surveys Tutorials **19**(4), 2432–2455 (2017)
30. Feng, J., Kong, L.: A fuzzy multi-objective genetic algorithm for QoS-based cloud service composition. In: Proc. of the 11th Int. Conf. on Semantics, Knowledge and Grids, pp. 202–206 (2015)
31. Ferrari, S., Frosio, I., Piuri, V., Borghese, N.A.: Automatic multiscale meshing through HRBF networks. IEEE Trans. on Instrumentation and Measurement **54**(4), 1463–1470 (2005)
32. Fortune: Amazon announces a security change that may help companies using AWS to avoid data breaches (2018). URL http://fortune.com/2018/11/16/amazon-secures-cloud-password-breach-safer

33. Fugue: Regions beyond regions: Global cloud infrastructure expansions (2016). URL https://www.fugue.co/blog/2016-04-12-regions-beyond-regions-global-cloud-infrastructure-expansions.html
34. Gonsalves, T., Itoh, K.: GA optimization of Petri net-modeled concurrent service systems. Applied Soft Computing **11**(5), 3929–3937 (2011)
35. Google: Google Cloud Platform (GCP) (2018). URL https://cloud.google.com/whitepapers
36. Guo, P., Xue, Z.: An adaptive PSO-based real-time workflow scheduling algorithm in cloud systems. In: Proc. of the 17th IEEE Int. Conf. on Communication Technology (ICCT), pp. 1932–1936 (2017)
37. Guzek, M., Bouvry, P., Talbi, E.: A survey of evolutionary computation for resource management of processing in cloud computing. IEEE Computational Intelligence Magazine **10**(2), 53–67 (2015)
38. Guzek, M., Pecero, J.E., Dorronsoro, B., Bouvry, P.: Multi-objective evolutionary algorithms for energy-aware scheduling on distributed computing systems. Applied Soft Computing **24**(C), 432–446 (2014)
39. Haykin, S.: Neural networks and learning machines. v. 10. Prentice Hall (2009)
40. Huang, W., Stokes, J.W.: MtNet: A multi-task neural network for dynamic malware classification. In: J. Caballero, U. Zurutuza, R.J. Rodríguez (eds.) Detection of Intrusions and Malware, and Vulnerability Assessment, pp. 399–418. Springer International Publishing (2016)
41. Huang, Y., Ma, X., Fan, X., Liu, J., Gong, W.: When deep learning meets edge computing. In: Proc. of the 2017 IEEE 25th Int. Conf. on Network Protocols (ICNP), pp. 1–2 (2017)
42. IBM: Data center energy efficiency (2017). URL https://www.ibm.com/ibm/environment/climate/datacenter_energy.shtml
43. IBM: IBM Cloud (2018). URL https://console.bluemix.net/docs/overview/ibm-cloud.html
44. IBM: Security in the IBM cloud (2018). URL https://www.ibm.com/cloud/security
45. IBM: Workload scheduler (2018). URL https://console.bluemix.net/catalog/services/workload-scheduler
46. IDG Enterprise: Cloud computing survey (2016). URL https://www.idgenterprise.com/resource/research/2016-idg-enterprise-cloud-computing-survey
47. Iturriaga, S., Nesmachnow, S., Dorronsoro, B., Talbi, E., Bouvry, P.: A parallel hybrid evolutionary algorithm for the optimization of broker virtual machines subletting in cloud systems. In: Proc. of the 8th Int. Conf. on P2P, Parallel, Grid, Cloud and Internet Computing, pp. 594–599 (2013)
48. Jain, A.K., Duin, R.P.W., Mao, J.: Statistical pattern recognition: A review. IEEE Trans. on Pattern Analysis and Machine Intelligence **22**(1), 4–37 (2000)
49. Jajodia, S., Kant, K., Samarati, P., Singhal, A., Swarup, V., Wang, C. (eds.): Secure Cloud Computing. Springer (2014)
50. Joshi, B., Vijayan, A.S., Joshi, B.K.: Securing cloud computing environment against DDoS attacks. In: Proc. of the 2012 Int. Conf. on Computer Communication and Informatics, pp. 1–5 (2012)
51. Kessaci, Y., Melab, N., Talbi, E.: A pareto-based genetic algorithm for optimized assignment of VM requests on a cloud brokering environment. In: Proc. of the IEEE Congress on Evolutionary Computation, pp. 2496–2503 (2013)
52. Kim, H.Y., Kim, J.M.: A load balancing scheme based on deep-learning in IoT. Cluster Computing **20**(1), 873–878 (2017)
53. Kolter, J.Z., Maloof, M.A.: Learning to detect and classify malicious executables in the wild. The Journal of Machine Learning Research **7**, 2721–2744 (2006)
54. Konar, A., Bhattacharya, D.: Time-Series Prediction and Applications: A Machine Intelligence Approach. Intelligent Systems Reference Library. Springer International Publishing (2017)
55. Kousiouris, G., Menychtas, A., Kyriazis, D., Konstanteli, K., Gogouvitis, S.V., Katsaros, G., Varvarigou, T.A.: Parametric design and performance analysis of a decoupled service-oriented prediction framework based on embedded numerical software. IEEE Trans. on Services Computing **6**(4), 511–524 (2013)

56. Kumar, J., Singh, A.K.: Workload prediction in cloud using artificial neural network and adaptive differential evolution. Future Generation Computer Systems **81**, 41–52 (2018)
57. Kumrai, T., Ota, K., Dong, M., Kishigami, J., Sung, D.K.: Multiobjective optimization in cloud brokering systems for connected internet of things. IEEE Internet of Things Journal **4**(2), 404–413 (2017)
58. L.D., D.B., Krishna, P.V.: Honey bee behavior inspired load balancing of tasks in cloud computing environments. Applied Soft Computing **13**(5), 2292–2303 (2013)
59. Lecun, Y., Bengio, Y., Hinton, G.: Deep learning. Nature **521**(7553), 436–444 (2015)
60. Li, L., Ota, K., Dong, M.: Deep learning for smart industry: Efficient manufacture inspection system with fog computing. IEEE Trans. on Industrial Informatics **14**(10), 4665–4673 (2018)
61. Lin, M., Wierman, A., Andrew, L.L.H., Thereska, E.: Dynamic right-sizing for power-proportional data centers. IEEE/ACM Trans. on Networking **21**(5), 1378–1391 (2013)
62. Linthicum, D.S.: Making sense of AI in public clouds. IEEE Cloud Computing **4**(6), 70–72 (2017)
63. Liu, L., Gu, S., Zhang, M., Fu, D.: A hybrid evolutionary algorithm for inter-cloud service composition. In: Proc. of the 9th Int. Conf. on Modelling, Identification and Control, pp. 482–487 (2017)
64. Liu, N., Li, Z., Xu, J., Xu, Z., Lin, S., Qiu, Q., Tang, J., Wang, Y.: A hierarchical framework of cloud resource allocation and power management using deep reinforcement learning. In: Proc. of the 2017 IEEE 37th Int. Conf. on Distributed Computing Systems (ICDCS), pp. 372–382 (2017)
65. Ma, X., Zhang, F., Chen, X., Shen, J.: Privacy preserving multi-party computation delegation for deep learning in cloud computing. Information Sciences **459**, 103–116 (2018)
66. Masdari, M., ValiKardan, S., Shahi, Z., Azar, S.I.: Towards workflow scheduling in cloud computing: A comprehensive analysis. Journal of Network and Computer Applications **66**, 64–82 (2016)
67. Microsoft: Azure load balancer overview (2018). URL https://docs.microsoft.com/en-us/azure/load-balancer/load-balancer-overview
68. Microsoft: Detecting fileless attacks with Azure security center (2018). URL https://azure.microsoft.com/en-us/blog/detecting-fileless-attacks-with-azure-security-center/
69. Microsoft: Microsoft Azure (2018). URL https://docs.microsoft.com/en-us/azure/security/security-white-papers
70. Mitra, B., Craswell, N.: Neural models for information retrieval. CoRR (2017)
71. Mozo, A., Ordozgoiti, B., Gomez-Canaval, S.: Forecasting short-term data center network traffic load with convolutional neural networks. PLoS One (2018)
72. Nae, V., Iosup, A., Prodan, R.: Dynamic resource provisioning in massively multiplayer online games. IEEE Trans. n Parallel and Distributed Systems **22**(3), 380–395 (2011)
73. Oracle: Oracle cloud workload migration (2017). URL https://www.infosys.com/Oracle/white-papers/Documents/practitioners-point-view.pdf
74. Oracle: Oracle Cloud (2018). URL https://docs.cloud.oracle.com
75. Pouyanfar, S., Sadiq, S., Yan, Y., Tian, H., Tao, Y., Reyes, M.P., Shyu, M.L., Chen, S.C., Iyengar, S.S.: A survey on deep learning: Algorithms, techniques, and applications. ACM Computing Surveys **51**(5), 92:1–92:36 (2018)
76. Qiu, J., Wu, Q., Ding, G., Xu, Y., Feng, S.: A survey of machine learning for big data processing. EURASIP Journal on Advances in Signal Processing **2016**(1) (2016)
77. Rack Solutions: The internet is on 75 million servers with 5 million terabytes of data (2014). URL https://dazeinfo.com/2014/04/22/internet-comprises-5-million-terabytes-data-weighs-much-grain-sand/
78. Rahman, S., Ahmed, T., Huynh, M., Tornatore, M., Mukherjee, B.: Auto-scaling vnfs using machine learning to improve qos and reduce cost. In: Proc. of the 2018 IEEE Int. Conf. on Communications (ICC), pp. 1–6 (2018)
79. Rahulamathavan, Y., Phan, R.C.., Veluru, S., Cumanan, K., Rajarajan, M.: Privacy-preserving multi-class support vector machine for outsourcing the data classification in cloud. IEEE Trans. on Dependable and Secure Computing **11**(5), 467–479 (2014)

80. Rana, O.: The costs of cloud migration. IEEE Cloud Computing **1**(1), 62–65 (2014)
81. Rasheduzzaman, M., Islam, M.A., Rahman, R.M.: Workload prediction on google cluster trace. Int. J. Grid High Perform. Comput. **6**(3), 34–52 (2014)
82. Saiyeda, A., Mir, M.A.: Cloud computing for deep learning analytics: A survey of current trends and challenges. International Journal of Advanced Research in Computer Science **8**(2) (2017)
83. Samarati, P., Jajodia, S.: Data security. In: J. Webster (ed.) Wiley Encyclopedia of Electrical and Electronics Engineering. John Wiley & Sons (1999)
84. Samreen, F., Elkhatib, Y., Rowe, M., Blair, G.S.: Daleel: Simplifying cloud instance selection using machine learning. In: Proc. of the 2016 IEEE/IFIP Network Operations and Management Symp. (NOMS), pp. 557–563 (2016)
85. Sanzo, P.D., Rughetti, D., Ciciani, B., Quaglia, F.: Auto-tuning of cloud-based in-memory transactional data grids via machine learning. In: Proc. of the 2012 2nd Symp. on Network Cloud Computing and Applications (NCCA), pp. 9–16 (2012)
86. Shabeera, T., Kumar, S.M., Salam, S.M., Krishnan, K.M.: Optimizing VM allocation and data placement for data-intensive applications in cloud using ACO metaheuristic algorithm. Engineering Science and Technology, an International Journal **20**(2), 616–628 (2017)
87. Sharifian, S., Motamedi, S.A., Akbari, M.K.: A predictive and probabilistic load-balancing algorithm for cluster-based web servers. Applied Soft Computing **11**(1), 970–981 (2011)
88. Simon, D.: Evolutionary Optimization Algorithms. Wiley (2013)
89. Sundararajan, K., Woodard, D.L.: Deep learning for biometrics: A survey. ACM Computing Surveys **51**(3), 65:1–65:34 (2018)
90. Symantec: Symantec enables security in the oracle cloud (2017). URL https://www.symantec.com/content/dam/symantec/docs/white-papers/symantec-enables-security-oracle-cloud-en-v1a.pdf
91. Synergy Research Group: Cloud revenues continue to grow by 50% as top four providers tighten grip on market (2018). URL https://globenewswire.com/news-release/2018/07/27/1543412/0/en/Cloud-Revenues-Continue-to-Grow-by-50-as-Top-Four-Providers-Tighten-Grip-on-Market.html
92. Tobiyama, S., Yamaguchi, Y., Shimada, H., Ikuse, T., Yagi, T.: Malware detection with deep neural network using process behavior. In: Proc. of the 2016 IEEE 40th Annual Computer Software and Applications Conf. (COMPSAC), vol. 2, pp. 577–582 (2016)
93. Trillas, E., Eciolaza, L.: Fuzzy logic: an introductory course for engineering students. Springer Publishing Company, Incorporated (2015)
94. Varghese, B., Buyya, R.: Next generation cloud computing: New trends and research directions. Future Generation Computer Systems **79**, 849–861 (2018)
95. Xu, J., Fortes, J.A.B.: Multi-objective virtual machine placement in virtualized data center environments. In: Proc. of the 2010 IEEE/ACM Int. Conf. on P2P, Parallel, Grid, Cloud and Internet Computing, pp. 179–188 (2010)
96. Xu, M., Tian, W., Buyya, R.: A survey on load balancing algorithms for virtual machines placement in cloud computing. Concurrency and Computation: Practice and Experience **29**(12) (2017)
97. ZDNet: Top cloud providers 2018 (2018). URL https://www.zdnet.com/article/cloud-providers-ranking-2018-how-aws-microsoft-google-cloud-platform-ibm-cloud-oracle-alibaba-stack/
98. Zhang, Y., Yao, J., Guan, H.: Intelligent cloud resource management with deep reinforcement learning. IEEE Cloud Computing **4**(6), 60–69 (2017)
99. Zhao, L., Sakr, S., Liu, A., Bouguettaya, A.: Cloud computing. In: Cloud Data Management, pp. 9–20. Springer International Publishing, Cham (2014)