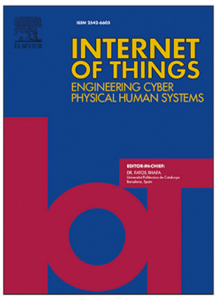
Journal Pre-proof



AI-based fog and edge computing: A systematic review, taxonomy and future directions

Sundas Iftikhar, Sukhpal Singh Gill, Chenghao Song, Minxian Xu, Mohammad Sadegh Aslanpour, Adel N. Toosi, Junhui Du, Huaming Wu, Shreya Ghosh, Deepraj Chowdhury, Muhammed Golec, Mohit Kumar, Ahmed M. Abdelmoniem, Felix Cuadrado, Blesson Varghese, Omer Rana, Schahram Dustdar, Steve Uhlig

PII: S2542-6605(22)00155-X
DOI: https://doi.org/10.1016/j.iot.2022.100674
Reference: IOT 100674

To appear in: *Internet of Things*

Received date : 18 November 2022
Revised date : 12 December 2022
Accepted date : 17 December 2022

Please cite this article as: S. Iftikhar, S.S. Gill, C. Song et al., AI-based fog and edge computing: A systematic review, taxonomy and future directions, *Internet of Things* (2022), doi: https://doi.org/10.1016/j.iot.2022.100674.

AI-based Fog and Edge Computing: A Systematic Review, Taxonomy and Future Directions

Sundas Iftikhar[a,b], Sukhpal Singh Gill*[a], Chenghao Song[c], Minxian Xu[c], Mohammad Sadegh Aslanpour[d,e], Adel N. Toosi[d], Junhui Du[f], Huaming Wu[f], Shreya Ghosh[g], Deepraj Chowdhury[h], Muhammed Golec[a,i], Mohit Kumar[j], Ahmed M. Abdelmoniem[a], Felix Cuadrado[l], Blesson Varghese[m], Omer Rana[n], Schahram Dustdar[o] and Steve Uhlig[a]

[a]*School of Electronic Engineering and Computer Science, Queen Mary University of London, London, UK,*
[b]*University of Kotli Azad Jammu & Kashmir, Kotli, Azad Kashmir, Pakistan,*
[c]*Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China,*
[d]*Department of Software Systems and Cybersecurity, Faculty of Information Technology, Monash University, Australia,*
[e]*CSIRO DATA61, Australia,*
[f]*Center for Applied Mathematics, Tianjin University, Tianjin, China,*
[g]*The Pennsylvania State University, Pennsylvania, USA,*
[h]*Department of Electronics & Communication Engineering, International Institute of Information Technology (IIIT), Naya Raipur, India,*
[i]*Abdullah Gül University, Kayseri, Turkey,*
[i]*Department of Information Technology, National Institute of Technology, Jalandhar, India,*
[l]*Technical University of Madrid (UPM), Spain,*
[m]*School of Computer Science, University of St Andrews, Scotland, UK ,*
[n]*School of Computer Science and Informatics, Cardiff University, Cardiff, UK,*
[o]*Distributed Systems Group, Vienna University of Technology, Vienna, Austria,*

ARTICLE INFO

Keywords:
Artificial Intelligence
Cloud Computing
Fog Computing
Edge Computing
Machine Learning
Internet of Things
Systematic Literature Review

ABSTRACT

Resource management in computing is a very challenging problem that involves making sequential decisions. Resource limitations, resource heterogeneity, dynamic and diverse nature of workload, and the unpredictability of fog/edge computing environments have made resource management even more challenging to be considered in the fog landscape. Recently Artificial Intelligence (AI) and Machine Learning (ML) based solutions are adopted to solve this problem. AI/ML methods with the capability to make sequential decisions like reinforcement learning seem most promising for these type of problems. But these algorithms come with their own challenges such as high variance, explainability, and online training. The continuously changing fog/edge environment dynamics require solutions that learn online, adopting changing computing environment. In this paper, we used standard review methodology to conduct this Systematic Literature Review (SLR) to analyze the role of AI/ML algorithms and the challenges in the applicability of these algorithms for resource management in fog/edge computing environments. Further, various machine learning, deep learning and reinforcement learning techniques for edge AI management have been discussed. Furthermore, we have presented the background and current status of AI/ML-based Fog/Edge Computing. Moreover, a taxonomy of AI/ML-based resource management techniques for fog/edge computing has been proposed and compared the existing techniques based on the proposed taxonomy. Finally, open challenges and promising future research directions have been identified and discussed in the area of AI/ML-based fog/edge computing.

*Corresponding author at: School of Electronic Engineering and Computer Science, Queen Mary University of London, London, E1 4NS, UK.
s.iftikhar@qmul.ac.uk (S. Iftikhar); s.s.gill@qmul.ac.uk (S.S. Gill*); ch.song@siat.ac.cn (C. Song); mx.xu@siat.ac.cn (M. Xu); mohammad.aslanpour@monash.edu (M.S. Aslanpour); adel.n.toosi@monash.edu (A.N. Toosi); dujunhui_0325@tju.edu.cn (J. Du); whming@tju.edu.cn (H. Wu); spg5897@psu.edu (S. Ghosh); deepraj19101@iiitnr.edu.in (D. Chowdhury); m.golec@qmul.ac.uk (M. Golec); kumarmohit@nitj.ac.in (M. Kumar); ahmed.sayed@qmul.ac.uk (A.M. Abdelmoniem); felix.cuadrado@upm.es (F. Cuadrado); blesson@st-andrews.ac.uk (B. Varghese); ranaof@cardiff.ac.uk (O. Rana); dustdar@dsg.tuwien.ac.at (S. Dustdar); steve.uhlig@qmul.ac.uk (S. Uhlig)
ORCID(s): 0000-0002-3913-0369 (S.S. Gill*)
[1]Sukhpal Singh Gill co-led this work with first author.

1. Introduction

Most modern web applications now follow the standard practice of tapping into the remote computing resources provided by cloud data centers [1]. Mobile phones, wearables, and other user devices, as well as sensors in a smart city or factory, all create data that is often sent to remote clouds for processing and storage [2]. Due to the likelihood of a rise in communication latencies when billions of devices are linked to the Internet, this computing architecture is impractical for the long term [3]. The increased communication latencies will negatively affect applications and lower the Quality of Service (QoS) [4]. Bringing computing resources nearer to end devices and sensors and employing them for data processing is an alternate computing strategy that can help with the aforementioned issue (even if only partially) [5]. This might lessen the load placed on the cloud and speed up communications. The recent fashion in computing research is to implement this concept by moving part of the processing power currently housed in huge data centers to the network's periphery, where it will be closer to end-users and sensors [6]. Internet of Things (IoT) devices such as routers, gateways, and switches may be equipped with computer resources, or specialised "micro" data centers may be built within public/private infrastructure for the ease of access and security [7]. *"Edge computing" refers to a computing model that takes advantage of network edge resources. "Fog computing" refers to a paradigm that employs both on-premises hardware and cloud services [8].* Edge resources differ from cloud resources in several ways [9, 10, 11]: (a) they are resource constrained, meaning they have fewer computational capabilities due to edge devices' smaller processors and lower power budgets; (b) they are heterogeneous, meaning that different processors use different configurations; and (c) their workloads adjustment and applications fight for them. Hence, one of the major difficulties in fog and edge computing is managing resources.

1.1. Resource Management Issues in Fog/Edge computing

In recent years, IoT applications (e.g. smart homes, self-driving cars, smart agriculture, smart healthcare) have improved people's quality of life [12]. The increase in IoT applications has also increased a number of IoT devices such as sensors, smart CCTV cameras, smart gadgets, and other smart devices. These IoT applications generate a massive amount of data [13]. According to a report by International Data Corporation, in 2025 data generation from IoT devices will reach 79.4 zettabytes [14]. Traditional cloud infrastructure is not designed to handle such a huge amount of data [15]. The large amount of data generated from the actuators, mobile devices and sensors, has incorporated latency, network bandwidth and security challenges to cloud infrastructure for time-sensitive applications [16]. To overcome these challenges, the emerging distributed computing paradigm "fog computing" and "edge computing" as an extension of Cloud computing has drawn the attention of the industrial and research community [17]. Fog/edge computing provides computing, network and storage services and control close to the data origin by combining distanced resources between cloud and end devices [18]. Though the resources in fog/edge are more limited in capability than cloud resources, they can play an important role in processing data for time-sensitive or real-time applications [19]. It enables location awareness, user mobility support, real-time interactions, low latency, high scalability, and interoperability that cloud-based systems could not support [20]. But the increase in IoT applications and limited resources in fog/edge computing environments has made efficient resource management very crucial.

1.2. Need of AI/ML for Fog/Edge computing

With the increase in the use of IoT and Machine Learning (ML), cloud and fog/edge workloads are becoming increasingly diverse and dynamic. The confluence of fog and AI for improvement in human quality of life necessitates the use of smart management of fog resources. In traditional cloud computing platforms, resource management is done using traditional heuristic approaches without considering diverse and dynamic workloads [21]. Most of these methods (e.g., Threshold-based method) are static heuristics configured offline to certain workload scenarios. They are not able to scale applications in and out at run time based on the pattern and behavior of workload [22]. The performance of heuristic methods can also be drastically downgraded when the system is scaled up. Resource contention is also a major problem in fog environments where co-located applications compete for shared resources in such policies and cause performance deterioration and Service Level Agreement (SLA) violation [23]. The shift of application structure from monolithic applications to miro-services and serverless has also increased the complexity [24]. Dependency in micro-services may cause Service Level Objective (SLO) violations due to communication costs and higher resource demands in fog computing.

The fast-rising diversity of workloads, the complexity of applications and the near optimum requirement of QoS parameters of some IoT applications in Fog/Cloud environments, motivate the utilization of AI/ML techniques to optimize their resource management policies [25]. AI and ML models could be used to model and predict application

and infrastructure level metrics that could also assist in task/resource orchestration by improving the quality of resource provisioning decisions [26]. Also, ML method can be directly used for resource management decisions for high accuracy and lower time overhead in large-scale systems [27]. ML algorithms e.g., Support Vector Machine (SVM), and polynomial regression, can be used to explore relations between performance metrics [1], the K-means algorithm can be used for the detection of abnormal system behaviors, Reinforcement learning models can be adopted for decision-making for resource provisioning, advanced Recurrent neural network can be used for the analysis of resource utilization or regression of application performance metrics and SVM can also be utilized for dependency analysis of application components [28].

1.2.1. Motivation

In the fog/edge computing context, AI/ML-based solutions have been employed for a variety of goals, including resource efficiency, load balancing, energy efficiency, SLA assurance, etc. Therefore in this article, we aim to investigate "AI for fog/edge computing" and its components for the realization of fog/edge-enabled AI.

- AI/ML-based resource management techniques have demonstrated potential for managing resources and deploying applications in cloud computing. Hence, aim to outline the evolution and principles of AI/ML-based resource management in fog/edge computing in recent studies.
- Existing survey papers do provide light on AI/ML-based resource management for fog/edge computing, but this area of study is rapidly growing as new AI/ML models are integrated. In order to uncover new research problems, trends, and potential future directions, a new Systematic Literature Review (SLR) of AI/ML-based resource management systems for fog/edge computing is required.

1.3. Related Surveys and Contributions

Many reviews/surveys have been conducted that discuss the role of AI/ML in fog/edge computing. Ghobaei-Arani *et al.* [12] reviewed solutions for resource management approaches in fog computing. They presented a taxonomy of resource management methods considering six dimensions, resource planning, load balancing, task offloading, resource allocation, resource provisioning, and application placement. They presented a thorough analysis of several case studies and their methodologies but focused on general approaches and partially discussed AI approaches. They did not provide any classification of AI-based solutions. Zhong *et al.* [1] presented a review of machine learning approaches for container orchestration issues from resource management. They proposed a taxonomy to classify current research by its common features. Duc *et al.* [29] investigated machine learning-based resource provisioning in joint edge-fog-cloud environments, and surveys technologies, mechanisms, and ML-based methods that can be used to improve the reliability of distributed applications in diverse and heterogeneous network environments. Casalicchio *et al.* [30] explored the problem of autonomic container orchestration and presented a taxonomy of container technology, container tools, and architecture, but they only provided a generalized discussion on container technology not specific to edge or fog. Another review [31] addressed the confluence of edge computing and AI. This work has two-dimensional agenda: the use of edge computing for AI and the use of AI for Edge. They only focused on computational offloading and mobility management with AI methods and only discussed a few AI-based works for these issues. Kansal *et al.* [32] presented a review of data-driven approaches for fog management issues. They are classified based on the technology used, QoS factors, and data-driven strategies. However, they generically reviewed all the data-driven techniques and do not present any classification or taxonomy of AI techniques.

Although existing survey articles provide new insights into AI/ML-based resource management for fog/edge computing, the research field is constantly expanding with the integration of new AI/ML models. Therefore, new reviews of AI/ML-based resource management approaches are needed to identify emerging research challenges and possible future directions. Further, none of the existing surveys have used the Systematic Literature Review (SLR) approach to conduct the survey. In this work, we followed a systematic review methodology as per the "Centre for Reviews and Dissemination (CRD) guidelines" given by Kitchenham [33] to conduct this review on AI/ML-based resource management in Fog/Edge computing. Table 1 compares the related surveys with our SLR based on important key parameters.

1.3.1. Our Contributions

The main contributions of this Systematic Literature Review (SLR) are summarized as follows:

- Review AI/ML approaches used for the realization of AI/ML for fog/edge computing.

Table 1
Comparison of related surveys with our Systematic Literature Review (SLR)

Work		[12]	[30]	[31]	[1]	[34]	[35]	[36]	[29]	[37]	[38]	[39]	[32]	Our SLR
Year		2020	2019	2020	2022	2020	2019	2018	2019	2021	2022	2021	2022	2022
Environment	IoT		✓				✓	✓	✓					✓
	Edge			✓					✓					✓
	Fog	✓		✓		✓	✓	✓		✓	✓	✓	✓	✓
	Cloud		✓		✓				✓					✓
AI Method	Machine learning				✓				✓	✓				✓
	Deep learning									✓		✓		✓
	Reinforcement learning									✓	✓		✓	✓
AI for Fog/Edge	Resource Discovery												✓	✓
	Resource Estimation	✓											✓	✓
	Application Placement	✓				✓				✓			✓	✓
	Resource Orchestration		✓		✓								✓	✓
	Resource Scheduling	✓											✓	✓
	Resource Provisioning	✓			✓		✓		✓				✓	✓
	Task offloading	✓		✓									✓	✓
	Load balancing	✓											✓	✓
Taxonomy		✓	✓	✓	✓				✓		✓			✓
Classification					✓			✓	✓	✓		✓		✓
Systematic Literature Review (SLR)														✓

- Offer a comprehensive literature review to discuss the background and current status of AI/ML-based resource management approaches in fog/edge computing environments.
- Propose a taxonomy of the most common AI algorithms used for resource management in fog/edge computing environments.
- Compare existing studies using various parameters related to identified categories through the proposed taxonomy.
- Identify open issues and future directions for the confluence of edge and AI as Edge AI.

1.4. Article Organization

The rest of this article is structured as illustrated in Fig. 1. In Section 2, the review methodology is described. Section 3 presents the background and current status of AI based resource management approaches, Section 4 gives a detailed review of AI/ML-based techniques used for resource management issues and their current status. Section 5 presents a taxonomy of frameworks and comparison analysis in AI-based edge and fog computing. Section 6 discusses result outcomes and Section 7 provides open issues and future directions. Finally, Section 8 concludes the paper.

2. Review Methodology

This work is a Systematic Literature Review (SLR) of AI/ML-based resource management in Fog/Edge computing. We followed a systematic review methodology as per the "Centre for Reviews and Dissemination (CRD) guidelines" given by Kitchenham [33] to collect the most relevant studies on this issue. The following steps are included in the process of reviewing this article: i) establishing the evaluation process; ii) describing the evaluation criteria; iii) creating the taxonomy; iv) performing the analysis; v) contrasting the different previous studies; vi) analyzing the finding and outcomes; and vii) emphasising promising research directions.

2.1. Planning the review

Creating research questions is the first step in designing the rules of evaluation. We used these carefully constructed queries to do additional searches across a variety of data sources. The review method identifies and accumulates relevant

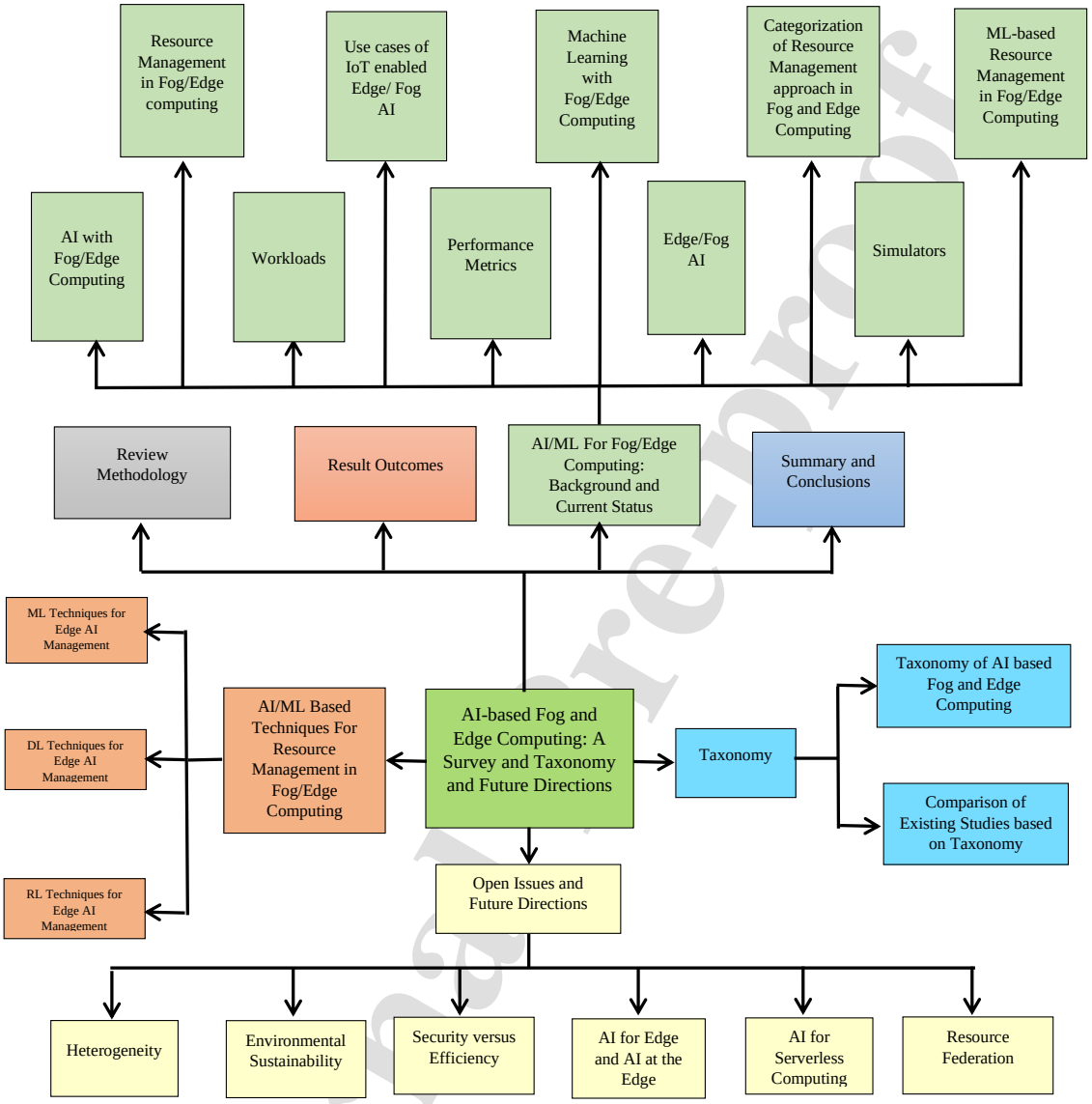


Figure 1: The Organization of this Systematic Literature Review (SLR)

data for the intended investigation. Articles are either taken into consideration or discarded heavily due to the evaluation procedure. The selection of this task by a single researcher might potentially instill bias in the study. This Systematic Literature Review was thus conducted by splitting among all of the contributors of this paper. Each author has written a document that outlines their thoughts on the review process and distributed it to other team members. Over a set period of time, this cycle has replicated itself. After much debate over several versions, the review guidelines have been completed. Several online databases have been combed thoroughly. Figure 2 represents the evaluation process.

2.2. Research questions

In order to better understand AI/ML for fog/edge computing, we plan to do a comprehensive overview of the field. Researchers may use the results of this study to have a better grasp of the state of AI/ML-based fog/edge computing and to pinpoint fruitful avenues for further investigation. Planning the review procedure requires the Research Questions

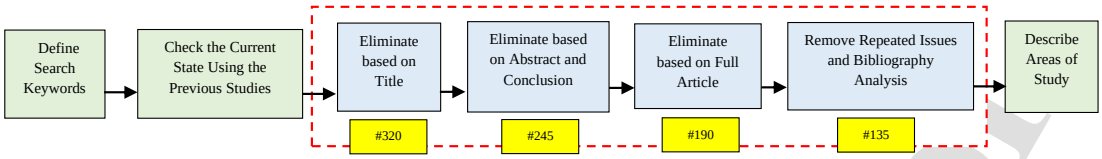


Figure 2: Process of Review Methodology

(RQ). Table 2 displays the research questions, the rationale behind them, and the relationships between various parts and subsections of our literature review to demonstrate how we are addressing these RQs using Systematic Literature Review (SLR).

2.3. Sources of information

A comprehensive search of electronic sources is essential for a thorough literature evaluation. In an effort to improve the probability of locating relevant research publications, we have selected the following collection of data sources:

- Wiley Interscience (www3.interscience.wiley.com)
- Springer (www.springerlink.com)
- ACM Digital Library (www.acm.org/dl)
- IEEE eXplore (ieeexplore.ieee.org)
- ScienceDirect (www.sciencedirect.com)
- Semantic Scholar (www.semanticscholar.org)

Additional Sources: In order to broaden our search for relevant research, we also consulted the following supplementary sources:

- Investigated the original sources included in the reference list.
- Technical Reports
- Edited Books and Text Books

2.4. Search criteria

The determined search method from various online sources is presented in Table 3. The research articles featured here were gathered using the most widely-used Internet resources in the field of AI/ML for fog/edge computing. ScienceDirect, IEEE Xplore, Springer, Taylor & Francis (T&F), ACM, Sage, Wiley, InderScience, and Google Scholar are only a few of the digital libraries from which the papers were retrieved. Finding relevant studies in the literature relies heavily on "Search string construction" and "Search keywords choice". Search terms like "fog computing" and "edge computing" and related terms like "Artificial Intelligence" and "Machine Learning" and "Deep Learning" revealed relevant items. Combining the keywords with the boolean operators AND and OR produced the final string for search. The following sequence has the following specified format:

```
[(Application placement) OR (Service placement) OR
(Task scheduling) OR (Container placement) ) OR
(VM placement) OR (Resource management) AND
((( Artificial Intelligence) OR (Machine Learning)) AND
(Challenges) OR (Metrics) OR (simulators) OR (Workload) OR
(Algorithms) OR (Methods))) AND (Edge computing) OR
(Fog computing) Or (Cloud computing)]
```

Table 2
Research questions, Motivation, Category and Mapping Sections

Sr. No.	Research question	Motivation	Category	Mapping Section
RQ1	What is the current status of AI/ML-based fog/edge computing?	The research question investigates the many different subareas within AI/ML-based fog/edge computing.	Current Status & Background and Result Outcomes	Section 3 and 6
RQ2	In fog/edge computing, what resource management methods are available that are based on AI and ML?	The purpose of this question is to delve into the numerous methods employed in either the simulation or real-time application of AI/ML-based fog/edge computing.	AI For Resource Management in Fog/Edge and Taxonomy	Section 4 and 5
RQ3	What are the most important subfields of AI/ML-powered fog/edge computing?	This question is useful for determining the nature of research that has been conducted utilising AI/ML-based fog/edge computing.	Current Status & Background and AI For Resource Management in Fog/Edge	Section 3 and 4
RQ4	Where are AI/ML-based fog/edge computing frameworks stand right now?	This inquiry probes the Multiple models for AI/ML-driven fog/edge computing that have been established by scholars for use in certain IoT use cases.	AI For Resource Management in Fog/Edge and Taxonomy	Section 4 and 5
RQ5	How can the efficiency of AI/ML-based fog/edge computing be measured, and what metrics are used for this purpose?	The effectiveness of AI/ML-based Resource Management Techniques for fog/edge computing is measured in terms of delay, cost, and power usage, among others, by the researchers.	Performance Metrics	Section 3.8
RQ6	What kinds of workloads are utilised to evaluate the efficacy of AI/ML-based fog/edge computing frameworks?	The survey identifies and mentions the workloads utilised by the fog/edge computing system.	Workloads	Section 3.10
RQ7	Which simulators are utilized for fog/edge computing that is based on AI/ML?	The paper identifies and discusses the simulators utilised in the fog/edge computing architecture for AI/ML-based Resource Management techniques.	Simulators	Section 3.9
RQ8	What are the most common applications of IoT-enabled Edge/Fog AI?	Use cases for IoT-enabled Edge/Fog AI are discovered and discussed in the paper.	Edge/Fog AI and Use-cases of IoT enabled Edge/Fog AI	Section 3.1 and 3.2
RQ9	What are ML/DL/RL, Online and Offline Learning Techniques for Edge AI Management?	Various techniques for Edge AI management are discovered and discussed in the paper.	ML with Fog/Edge and ML-based Resource Management	Section 3.4 and 3.7
RQ10	What are the techniques for Edge AI Management?	Various Deep Learning/Reinforcement Learning techniques for Edge AI management are discovered and discussed in the paper.	AI For Resource Management in Fog/Edge	Section 4
RQ11	How will AI and machine learning impact fog and edge computing in the future?	Finding out where fog/edge computing research is headed and what problems remain unanswered is crucial.	Open Challenges and Research Directions	Section 7

Firstly, we constructed a search query based on the formulated research questions in Table 2. Table 3 details the evaluation process's search strings.

2.5. Inclusion and exclusion criteria

AI/ML-based Fog/edge computing is a relatively new area of study, and only a small number of papers have addressed the key questions surrounding them prior to 2015. As a result, the number of articles covering the topic before 2015 was quite low. Figure 2 displays the selection procedure of research papers from the Internet and digital library databases. The aforementioned search terms and string combinations were utilized to narrow the available databases down to the most pertinent articles. Starting with publications that were not peer-reviewed or indexed by ISI, 320+ papers were chosen for the first phase. To find quality publications, a research screening method has been done to exclude brief publications, non-peer-reviewed papers, low quality book chapters, and low-quality studies that weren't

Table 3
Search strings for e-resources

Sr. No.	e-resource	Search String	Dates	Source Type	Subjects
1	ieeexplore.ieee.org	Abstract: Artificial Intelligence or Machine Learning for Fog Computing or Edge Computing	2016 - 2022	Conferences, Journals, Magazines and Transactions	Fog Computing, Edge Computing, Machine Learning, Deep Learning, Artificial Intelligence
2	www.springerlink.com	Abstract: Artificial Intelligence or Machine Learning for Fog Computing, Edge Computing	2016 - 2022	Conferences, Journals and Magazines	Fog Computing, AI, ML, DRL, RL, Edge Computing
3	www.sciencedirect.com	Abstract: Artificial Intelligence or Machine Learning for Fog Computing, Edge Computing	2016 - 2022	All sources	Fog Computing, Deep Learning, Artificial Intelligence, Edge Computing
4	www.onlinelibrary.wiley.com/	Abstract: Artificial Intelligence or Machine Learning for Fog Computing	2016 - 2022	Conferences, Journals, Magazines and Transactions	Fog Computing, Edge Computing, Machine Learning, Deep Learning, Artificial Intelligence
5	www.acm.org/dl	Abstract: Article Title: Fog, Full Text/Abstract: Artificial Intelligence or Machine Learning for fog or edge	2016 - 2022	Conferences, Journals, Magazines and Transactions	AI/ML, Fog, Cloud, Edge
6	www.taylorandfrancis.com/	Abstract: Artificial Intelligence or Machine Learning for Fog/Edge Computing	2016 - 2022	Conferences and Journals	Edge, Fog, AI, ML, DRL
7	www.inderscience.com/	Abstract: Artificial Intelligence or Machine Learning for Fog Computing	2016 - 2022	Journals	All Subjects
8	www.semanticscholar.org	Abstract: AI/ML for Fog/Edge Computing	2016 - 2022	arXiv Preprints	Fog Computing, Edge Computing, Machine Learning, Deep Learning, Artificial Intelligence
9	Other Publishers	Article Title: Fog, Full Text/Abstract: AI/Ml for fog or edge	2016 - 2022	All sources	Edge, Fog, AI, ML, DRL

capable of delivering any technical knowledge and scientific argument. By the end of the process, 135 articles from prestigious journals and conferences had been hand-picked for this evaluation. In Section 5, the suggested taxonomy is explained alongside an analysis of each work that fits into it.

The elimination of research was performed using the following criteria to pick the rigorous quality publications:

- Neither the journal nor the conference are indexed.
- The articles present any survey and analysis work.
- These are the documents that were not written using the English language.
- Works that do not undergo a rigorous peer review procedure.

2.6. Quality assessment

There are several research publications on AI/ML for fog/edge computing in a wide variety of journals and proceedings from conferences. After applying the exclusion and inclusion criteria, we conducted a quality evaluation of the selected papers to choose the most relevant ones for further consideration. To evaluate the studies' overall quality, we checked them against critical factors such as objectivity, internal consistency, and bias using the CRD

recommendations [33]. We have established quality evaluation forms as presented in **Appendix A** to evaluate high-quality research papers for this systematic review of the literature. We've asked both broad, exploratory questions and in-depth, exploratory ones. Preliminary Examining questions are a helpful tool for locating high-level research publications that are associated with AI/ML in Fog/Edge Computing. In addition, specific questions are used to choose the research papers that are the most pertinent to the primary context of AI/ML in Fog/Edge Computing.

2.7. Data extraction

The methodology for extracting data from the 135 research papers included in this analysis was detailed in **Appendix B**. Initiating the data-gathering process inspired us to create this data extraction form in order to answer the research questions. Our carefully stated selection criteria allowed us to identify the best works on AI/ML for fog/edge computing from a wide range of prestigious journals and conferences as listed in **Appendix C**. In addition, we have reached out to a number of authors in order to collect the necessary information regarding scholarly works. In this SLR, we used this procedure to retrieve the data:

- A set of authors read through all 135 publications to collect the necessary information.
- Other authors used random samples to verify the accuracy of data collection.
- Any issue that arose throughout the cross-checking procedure was discussed and settled in a number of meetings.

2.8. Acronyms

Abbreviations utilized in the systematic literature evaluation are given in **Appendix D**.

3. AI/ML For Fog/Edge Computing: Background and Current Status

In this section, we discuss the background concepts, including AI and ML with Fog/Edge Computing. Further, this section presents other concepts such as Resource Management in Fog/Edge computing, Categorization of Resource Management approach in Fog and Edge computing, IoT Applications, Performance Metrics, workloads and simulators. Figure 3 represents a broad taxonomy of AI/ML for Fog/Edge Computing.

3.1. Edge/Fog AI

The emerging computing model named fog and edge computing can alleviate the problem of bringing the computational resources closer to the end user. These computing models offered the services to several latency-sensitive IoT applications such as vehicular networks, agriculture, healthcare, smart home, and transportation system [40], where cloud models fall behind in handling the services with minimum response time [41]. The fog/edge paradigm supports low latency, high mobility, and interoperability with resource constraints for IoT applications [42]. The contemporary research trend resides in the decentralization of resources towards the edge of the network. In contrast to cloud resources, edge resources need distinctive managerial techniques because of underlying heterogeneous resources, dynamic workload, scalable data centers, and last but not least, unpredictability, fluctuating interactions and multi-tenancy across end users [43]. The dynamic workloads make the process even more complex when real-time applications are competing for limited resources [44]. Failure recovery, data redundancy, high cost, power consumption, and privacy are still issues with the emerging computing paradigm, it necessitates the management of fog/edge resources is considered one of the significant challenges and needs to be addressed by the intelligent solution to improve the performance metrics and resolve the mentioned issues [45]. Resource provisioning, task offloading, resource scheduling and allocation, service placement, and load balancing are the components of resource management [46]. Each component of resource management and its related issues are discussed briefly.

3.1.1. Resource provisioning

Resource provisioning is defined as selection, deployment, and run time management of software and hardware resources for the efficient performance of applications. There are fluctuations in IoT devices' workload that leads to the issue of over and under provisioning. In the case of overprovisioning, a greater number of resources are allocated as compared with the required IoT workload, and IoT users must pay more for the services used [47]. In case of under provisioning, a smaller number of resources are allocated for IoT services, as per the requirement of IoT workload and it increases the possibility of SLA violations [48]. Hence, an efficient mechanism is needed to overcome the mentioned challenges and provide the resources based on the service demands.

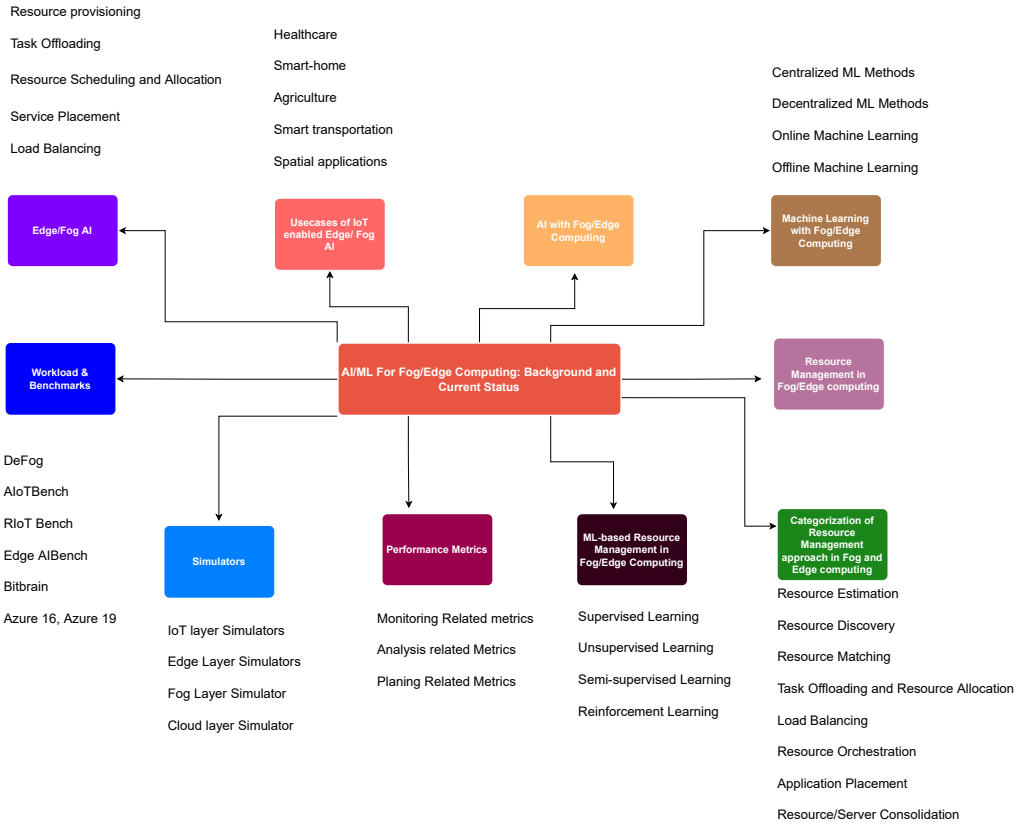


Figure 3: Taxonomy of AI/ML for Fog/Edge Computing

3.1.2. Task Offloading

It is problematic to take the offloading decision at runtime due to the complex architecture of fog and edge networks with resource constraints and allocate the best possible resource (cloud or fog) for computation-intensive tasks. The most common applications it supports are virtual reality, vehicular networks, and multimedia delivery [12]. We required an intelligent agent to decide, where the IoT devices-based workload will be processed and return the results within the deadline. The offloading decision are depending upon several factors like types of workloads, deadlines, priority, communication link, the capacity of fog nodes, and IoT devices. The main aim is to, utilize the link, and improve the latency and power consumption.

3.1.3. Resource Scheduling and Allocation

The number of fog/edge nodes are available to process the IoT requests, but required an efficient scheduling technique that will search the optimal resources for the upcoming workload and execute it within the deadline [49]. The scheduling of IoT service requests with objective function over the heterogeneous fog/edge nodes belongs to the class of NP-Complete and it is difficult to find the exact solution to the problem [50]. Resource scheduling and allocation is a different problem in the edge/fog paradigm with additional entities as compared with the cloud paradigm, and becomes a double mapping problem under-provisioning and IoT services demands [51].

3.1.4. Service Placement

The objective of service placement is to look at the optimum resources for IoT services, and deploy over the virtualized edge/fog nodes to enhance the QoS metrics, while maintaining the SLA [37]. An IoT application can be mapped with more than one fog/edge node or multiple services can be placed over a single fog node, depending upon the requirement of services and computation capacity of resources.

3.1.5. Load Balancing

It is one of the vital issues to distribute the workload over the virtualized fog nodes in balancing mode and avoid the possibility of over or underutilization of resources [52]. The goal of load balancing in edge/fog computing is to reduce the response time for latency-sensitive applications and address the challenges like network delay, high waiting time, and scalability to improve the system performance with potential solutions [53].

There are many resource management solutions are existed for the cloud paradigm, but cannot apply the same for fog/edge computing due to different network conditions, and characteristics, more distributed infrastructure, and processing capabilities of nodes [50, 53]. Hence, it is more challenging to address the issues of resource management in the fog/edge platform, as compared to the cloud platform [54]. Several researchers are working in this direction to manage the resources of a heterogeneous network, accurate offloading decisions, optimal provisioning, intelligent scheduling techniques, best-effort service placement and efficient workload sharing for load balancing, but no one has explored all the mentioned challenges entirely despite its importance for real-time applications, hence it opened the door for the new researchers to propose a novel solution for the existing issues.

3.2. Usecases of IoT enabled Edge/Fog AI

IoT has attracted significant research interests from both industry and academia and facilitates varied novel applications, including smart home, surveillance, smart healthcare and so on. Here, we present a brief summarization of different types of IoT applications.

3.2.1. Healthcare

IoT solutions have been considered and deployed for health management systems by efficiently tracing agents (patients, medical practitioners, medical resources), automatic data sensing and authentication and it is defined as *Internet of Health Things (IoHT) [55]*. IoHT technology has redefined the healthcare system by health monitoring of patients anytime and anywhere for post-discharge care, elderly health management and several other emergency situations like pandemic [56]. Wearable sensor is one of the major components in IoHT where health-related parameters are collected in different time intervals and processed for smart e-healthcare applications [57, 58]. A framework with IoT-based wearable sensors coupled with machine learning methods has been proposed for monitoring sport's person health conditions by collecting health parameters and exercise traces [59]. Carlos *et al.* [60] presented an IoHT-based deep learning framework for medical image (cerebral vascular accident image, lung nodule and skin images) classifications [60]. Another work by Ray *et al.* [61] designed a prototype of a cost-effective and low-power sensor system that is conducive to monitoring real-time intravenous (IV) fluid bag levels in e-healthcare applications. A collaborative edge-IoT framework, named RESCUE is proposed in [62] for provisioning healthcare services specifically in exigency time by collecting patient's location, and health condition and predicting the route of nearby healthcare centers. The framework also devises latency-aware and power-aware frameworks using IoT devices. Several research works have been carried out to mitigate COVID-19 by leveraging IoT-based solutions using AI/ML [63, 64]. Khan *et al.* [65] present *DCA-IoMT*, a location-aware knowledge-graph-based recommendation framework for an alert generation against COVID-19. FairHealth, an Internet of Medical Things (IoMT) framework is proposed in [66], where the fairness-aware resource scheduling method is deployed in 5G edge healthcare. Another imperative issue in the healthcare domain is the privacy aspect since such collected health data is sensitive in nature [67]. To mitigate such issues, a secure IoMT framework is presented leveraging blockchain [68]. In particular, when IoMT devices send data using a patient's Personal Digital Assistant (PDA), the data is transacted on the blockchain by the cloud server. Similarly, in the context of COVID-19, a blockchain-based privacy-preserving algorithm is proposed by Lv *et al.* [69] for contact tracing. The authors investigate several practical challenges including protecting data security and location privacy, dynamically and effectively deploying short-range communication IoT for activity-tracking and location-based services in large areas. The utility of IoT is explored for vaccine supply chain distribution in India [70].

3.2.2. Smart-home

IoT solutions can provision smart home services including automatic control of domestic appliances, alarm generation, security controls and developing an Internet-connected system. Gavrila *et al.* [71] present an IoT-based framework for seamless integration with a Hybrid broadcast broadband TV-enabled television set in a smart home environment for a better user experience. A multi-objective and smart residential load management framework is presented for energy management in smart-homes [72]. Specifically, an IoT based controller manages the home loads and generates alerts if any malfunction in the household loads is detected. In the smart-home context, cyberattacks

cause potential harm to the occupants and compromise their safety. In this regard, Yamauchi *et al.* [73] devised a novel method to detect such attacks by learning occupants' behavior as sequences of events such as the operation of home IoT devices and activities along with environmental variables (temperature, humidity, time of the day). The method compares learned sequences and current sequences when an operation command is activated, and an anomaly is detected. Kratos+, a multi-user and multi-device-aware access control mechanism is proposed in a similar context for allowing smart home users to specify the access control demands [74]. Li *et al.* [75] present a human pose forecasting system for smart homes leveraging graph convolutional neural network on the IoT edge for online learning. IoT Meta-Control Firewall (IMCF+) is proposed to mitigate energy consumption and CO2 emission issues while also maintaining user comfort [76].

3.2.3. Agriculture

IoT has brought dramatic improvements in agricultural production by enhancing the quality of agricultural products, reducing labor costs, and effective farm management [77]. Alahi *et al.* [78] design a smart nitrate sensor that monitors nitrate concentrations in ground and surface water. The system is supported by WiFi-based IoT that can send data directly to an IoT-based web server and serves as a distributed monitoring system [79]. A cyber-physical system for crop evapotranspiration estimation is proposed [80]. A gradient-boosting decision tree along with a fuzzy granulation method is used on IoT data from Xi'an Fruit Technology Promotion Center in Shaanxi Province, China for cherry tree evapotranspiration estimation and the proposed system achieved promising accuracy [81]. The continuous monitoring of crop growth is one of the most important aspects of precision agriculture. Bauer *et al.* [82] design a complementary framework for low-cost crop sensing leveraging temporal variations of the signal strength of low-power IoT radio communication [83]. Multidimensional feature compensation residual neural network (MDFC-ResNet) framework [84] identifies fine-grained crop disease using IoT technology and deep learning method.

3.2.4. Smart transportation

IoT demonstrates a promising future in Intelligent Transportation Systems (ITS) by collecting, analyzing traffic/mobility-related data and developing a smart, safe, reliable and sustainable ITS [85]. A smart parking surveillance system (detecting parking occupancy) is proposed by using edge computing and real-time video feed [86]. Bansal *et al.* [87] propose *DeepBus* for identifying surface irregularities (e.g., potholes) on roads using IoT sensor and machine learning methods. The system centrally hosts a map and alerts users and authorities regarding pothole locations. Philip *et al.* [88] designed an IoT-based smart traffic control system where a group of self-driving cars interact with road-side units and independently decide their lane velocities. IoT-based energy efficient ITS framework is presented that can reduce energy consumption, noise pollution, waiting time and greenhouse gas emissions in smart city environment [89]. Wan *et al.* [90] proposed a framework consisting IoTs of vehicles for vehicle number estimation which in turn helps in vehicle localization. A predictive framework is designed for forecasting the parking space occupancy leveraging deep learning-based ensemble technique [91] in IoT environment. The system specifically reduces the search time for parking and the optimization of the flow of cars helps in better traffic management in congested areas of a city.

3.2.5. Spatial applications

Internet of Spatial Things (IoST) integrates spatial or location information in the core IoT architecture to facilitate location-aware services [92, 93]. Ghosh *et al.* [94] presented a mobility-aware IoST framework for time-critical applications (e.g., ambulance service, disaster relief) for predicting optimal paths with less delay. Koh *et al.* [95] proposed a new location spoofing detection algorithm that can be used for spatial tagging and location-based services in an IoT environment. A spatial-data driven IoT framework, *STOPPAGE* is developed for predicting COVID-19 hotspot zones and efficient medical resource management in varied regions [96].

3.3. AI with Fog/Edge Computing

IoT is a communication network created by objects that can connect to the Internet and communicate with each other [97]. It has started to be used everywhere, from healthcare applications to military applications, and it is estimated that the number of IoT devices will reach approximately 30 billion by 2030 [98]. Along with the vertical increase in the number of IoTs, the amount of data that needs to be processed and produced by sensors has reached gigantic proportions. Processing this data in the cloud seems like a logical solution at first because of its advantages, such as high processing power and storage capacity [99]. However, problems such as [100] latency may occur in time-sensitive IoT applications such as instant patient follow-up.

Fog computing can be seen as an inspiring development to solve problems such as latency, power consumption, and network traffic in Cloud-based IoT systems [101]. Unlike cloud data centers, Fog nodes are located close to the IoT layer. Thus, execution time and bandwidth issues can be reduced [102]. On the other hand, fog nodes do not consist of devices with powerful processing power and large storage ability such as cloud data centers [12]. Therefore, one of the difficulties that need to be solved in fog computing is resource management, which consists of subheadings such as resource scheduling, task offloading and resource provisioning [9].

Resource management issue for Edge/Fog AI is addressed using diverse techniques. One of these methods is AI-based techniques that have been gaining popularity recently. AI-based techniques used to solve resource management problems in Fog/Edge computing can be summarized as Deep Learning (DL), Machine Learning (ML), Reinforcement Learning (RL), and Deep Reinforcement Learning (DRL). AI-based techniques are very effective in dynamic resource scheduling [32]. In particular, DRL has been shown to be very successful in dynamic complex problems and dynamic task offloading [32]. In addition, AI-based techniques such as neural networking and RL were found to be more popular in resource estimation than mathematical models [103].

3.4. Machine Learning with Fog/Edge Computing

AI and ML became an integral part of everyday application decision-making. It is used by recommender systems for tech giants such as Google, Amazon, Netflix, and Facebook and in more complicated use cases such as self-driving cars [104], earthquake prediction [105], and smart healthcare [106]. Due to the abundance of data sources at the edge, Fog/Edge computing received increasing attention as an enabler of Machine Learning methods. In this section, we examine Centralized vs Decentralized ML Methods and Online vs Offline ML for fog/edge computing.

3.4.1. Centralized ML Methods

AI and ML models feed on a tremendous amount of data generated by thousands to millions of mobile and IoT devices. Typically, these devices continuously stream the generated data into the cloud applications to be stored for later processing and analysis. These data are analyzed to extract certain features to help train AI/ML models. These models are trained on high-performance servers residing in the data centers of the cloud. Google Cloud, Microsoft Azure, Amazon AWS are the most common providers for ML-as-a-service where models can be trained on large amounts of data at scale. The interactions between the various services in the Fog/Edge are another source of training data that can be leveraged to enable more intelligent ML-based applications to be deployed and enhance the service for the users. However, the major concern with this setting is the security and privacy of the collected data used for training which may contain private and sensitive information. Other major problems for centralized Fog/Edge-based ML methods are latency and communication transfer costs [107, 108, 109].

There are many centralized learning methods for the purposes of workload prediction to aid with the resource allocation problem in literature [110, 111, 112, 113]. Wang *et al.* [111] proposed a feasible solution for edge cloud resource allocation over time based on an online algorithm to solve sub-problems with logarithmic objectives. The algorithm is shown to achieve a parameterized competitive ratio, without requiring any a priori knowledge of the resource price or the user mobility. The results with real-world and synthetic data confirm the effectiveness of the proposed algorithm. Rosendo *et al.* [112] provided an overview of the main state-of-the-art libraries and frameworks for ML and data analytics on the Edge-to-Cloud Continuum. This work also covers the main simulation, emulation, deployment systems, and testbeds. In addition, a holistic understanding of the performance optimization of applications and efficient deployment of AI/ML workflows is given. Nguyen *et al.* [113] proposed a market-based resource allocation framework in which the services act as buyers and fog resources act as divisible goods in the market. The aim is to compute a Market Equilibrium (ME) solution at which every service obtains its favorite resource bundle under the budget constraint, while the system achieves high resource utilization. The work discusses both centralized and privacy-preserving distributed solutions.

3.4.2. Decentralized ML Methods

Centralized learning (CL) for learning ML models is becoming obsolete because it requires the collection of decentralized user data imposing security and privacy risks and expensive data transfer [114, 115, 116, 117, 118]. Hence, decentralized paradigms are being explored as alternatives. Several techniques leverage decentralized learning methods for the purposes of workload prediction to aid with the resource allocation problem [119, 120, 121].

- **Federated Learning (FL):** In FL architecture, the learners are end-user devices such as smartphones, sensors, or IoT devices; training data is owned and stored at these devices; the learners train a global model collaboratively

with the assistance of a centralized FL (or aggregation) server [114, 117, 118, 122, 123]. As described in [114, 118, 122], the training of the global model occurs over a series of rounds until the model converges to a satisfactory accuracy. In each round, a few clients are sampled to update the model and a new model is produced. But, due to the server's central role, FL faces challenges of synchronization, reliability, and expensive communication [116]. At the start of each round, the server waits for available devices to check-in. The server selects a subset of these devices which meet certain conditions, such as being idle and connected to WiFi and a power source. Then, the server sends the global model along with the necessary configurations (i.e., hyper-parameter settings) to the selected clients. The learners perform the same number of local optimization steps as set by the server. Then, the learners send their updated models (or the delta) to the server. Finally, the server aggregates, with the global model, the model updates sent by the clients, and then checkpoints the new global model to the local storage [122]. One of the main challenges in FL use cases is the heterogeneity of the environment [114, 116] which is studied and addressed by several works [117, 118, 123].

- **Decentralized Learning (DL):** It is an alternative approach for training common models on decentralized data, typically in environments consisting of edge devices [115, 124]. In DL, the learners, without centralized coordination, engage in the learning process to train a model tailored to their common tasks and coordinate among themselves via peer-to-peer communication. Thus, device groups can train a common model while each device preserves its data. However, due to a lack of central coordination, the devices need to be available at the same time to iterate over the training process in a lock-step fashion, causing training to be as fast as the slowest device. This hinders the scalability and efficiency as devices can not train at their own pace without being held back by slow learners [115, 116].

There has been recent interest in techniques that leverage the non-conventional decentralized learning methods for the purposes of workload prediction to aid with the resource allocation problem [119, 120, 121]. Zarandi *et al.* [119] provides an optimization of the offloading decisions, computation resource allocation, and transmit power allocation for Edge IoT networks. The problem is presented as a multi-agent Distributed Deep Reinforcement Learning (DDRL) problem which is addressed via double deep Q-network (DDQN), where the actions are offloading decisions. Then, federated deep learning (FDL) is used to enhance the learning speed of IoT devices (agents) by creating a context for cooperation between agents with minimal impact on their privacy. Fantacci *et al.* [120] applies FL to train models for demand prediction. The proposed method achieves high accuracy levels on the predicted application demand via aggregating the feedback received from the user models. Chen *et al.* [121] propose a two-timescale federated deep reinforcement learning based on Deep Deterministic Policy Gradient (DDPG) to solve the joint optimization problem of task offloading and resource allocation to minimize the energy consumption of all IoT devices subject to delay threshold and limited resources. The simulation results show that the proposed algorithm can greatly reduce the energy consumption of all IoT devices.

3.4.3. Online Machine Learning

One of the design options used when modeling ML method is Online ML. In this model, the learning algorithm is constantly updated using new data [125]. Therefore, real-time data must be used in scenarios where Online ML is used for fog/edge computing. An example is models that predict the stock market [126]. Figure 4 shows the working scheme of Online ML [126]. ML parameters are updated by being trained by a new set of data each time. The learning step continues as new data comes in, and this process is quite fast and inexpensive. Online ML can be a suitable design option for scenarios where data flow is intense and constantly changing.

3.4.4. Offline Machine Learning

Unlike Online ML, there is no continuous data flow in Offline ML or Batch Learning. The ML model is trained using a certain number of data. After the model is trained, the test set performance is checked. If the test set performance is good enough, the learning phase ends. In case the model needs to be trained using new data, old and new data are used together. Figure 5 gives the working diagram of Offline ML [126]. Compared to online ML, the amount of data used to train the model is larger. Therefore, it is obvious that more CPU and RAM will be needed to train a model in Offline ML. In addition, with a large amount of data, it will take longer to train the models. In Fog/Edge Computing, offline ML methods are often used to solve offloading problems.

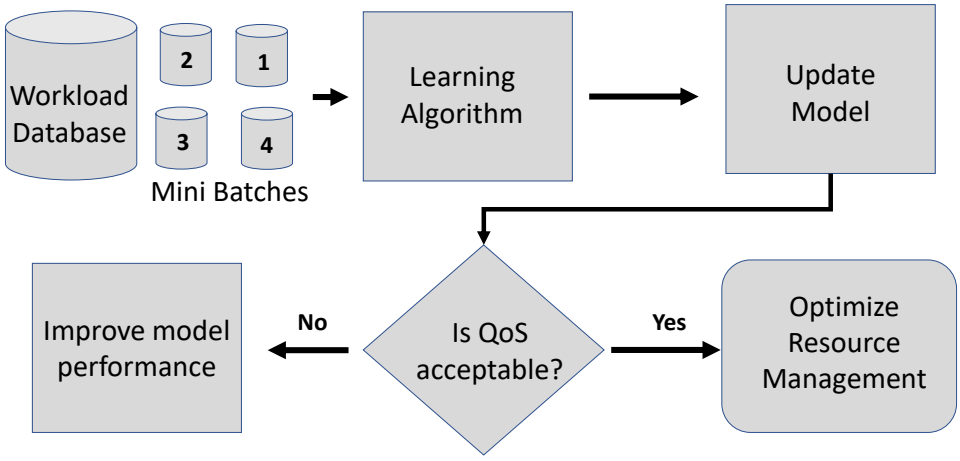


Figure 4: Online Machine Learning General Scheme for Fog/Edge Computing

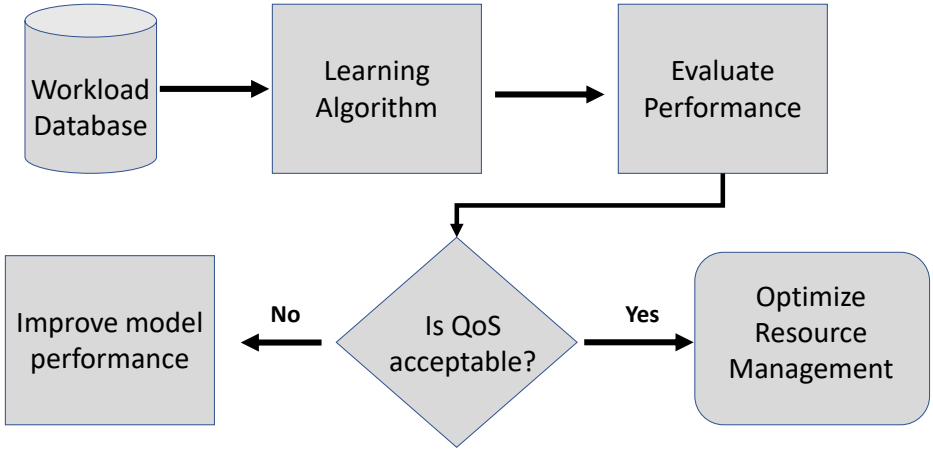


Figure 5: Offline Machine Learning General Scheme for Fog/Edge Computing

3.5. Resource Management in Fog/Edge computing

This computing architecture is not sustainable in the long run because of the expected increase in communication latencies when billions of devices are connected to the Internet. Application performance will suffer and QoS may drop as a result of longer connection delays [127]. An alternative computing method that can aid with this problem is to bring computer resources closer to end devices and sensors and use them for data processing. Communications might be sped up and cloud resources used could be reduced [128]. In recent years, there has been a trend in Computer Science to put this theory into practice by relocating some of the computing capacity now located in massive data centers to the network's perimeter, making it more accessible to end-users and sensors [129]. The Internet's routers, gateways, and switches may have access to computing power, or "micro" data centers may be set up in existing public and private networks for convenience and safety. Computing models that take advantage of network edge resources are known as "edge computing". Fog computing is the practice of combining local hardware with remote cloud resources.

Edge resources are distinct from cloud resources in that they are resource limited. This means that they have less computing capability than cloud resources because of the smaller processing units and reduced energy constraints of edge devices. They also employ various configurations for different CPUs, making them heterogeneous [130].

3.6. Categorization of Resource Management Approach in Fog and Edge computing

For the edge computing paradigm and the fog computing paradigm [131], the common denominator of the two is to sink the computing resources in the cloud to the user side, and provide better services for those user devices that do not have enough resources at a lower latency and energy consumption. To do this, we need to offload task data or place applications on another device or multiple devices, these devices usually have more computing resources or fewer energy constraints than the user device [132]. Generally, resource management is closely related to task offloading, in order to make better offloading decisions, we need to understand in detail the different resources in fog computing and edge computing scenarios, and these are all provided by resource management technologies [133]. For example, the estimation, discovery, and matching for resources can be used to make offload decisions, while resource allocation techniques can be used to perform offload decisions, and load balancing and resource orchestration or consolidation are designed to improve resource utilization and speed up response across the system after offloading tasks [134]. All in all, a better approach to resource management is to better offload task data or application placement to better serve users. Figure 6 shows the flow of resource management approaches in the edge and fog continuum for realizing Edge AI.

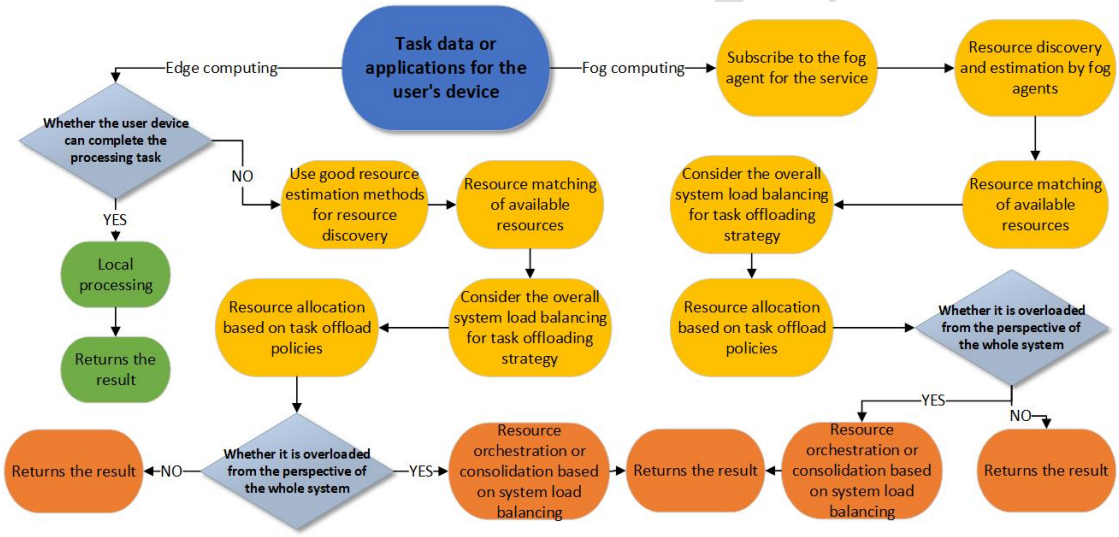


Figure 6: The flow chart of resource management in fog and edge computing.

3.6.1. Resource Estimation

Our estimates of resources under the two computational paradigms focus on the following five aspects [135]: computational resources [136] (e.g., CPU computational frequency, the number of CPU cycles required for computing one bit data, etc.), communication resources [137] (e.g., spectrum resources under Frequency division multiple access (FDMA) [138], length of time allocated per user under Time-division multiple access (TDMA) [139], etc.), storage resources (e.g., memory for devices, flash memory, etc.), data resources (e.g., some popular content), energy resources (e.g., battery power, virtual energy queues, etc.).

3.6.2. Resource Discovery

For resource discovery, it's mostly about discovering which resources are available, where are they located, and how long can they be used (especially devices with batteries). Regarding the implementation of resource discovery, there are two main ways: centralized and distributed. Centralized [140, 141] refers to selecting a device as a Cluster Header (CH) to record resources on other devices in a cluster of many devices, or setting up a central resource agent as

the CH to record resource information for other devices. Once the user has a need, the user sends a message requesting the service to a nearby node, and then the requested device node will check whether it meets the user's needs, if it does not meet, the node will send the user's request packet to the CH of the cluster where the node is located, and then the CH will retrieve the resource record table on it to find a node that meets the user's needs for the user; Distributed [142, 143] refers to the fact that there is no CH to record the resources of other devices, when there is user demand, send a request message to the surrounding agent nodes, and then the requested node checks whether it meets the user needs, if not, the agent node (or the mobile device itself) sends resource request packets directly to all surrounding nodes by broadcasting to "discover" the required resources.

3.6.3. Resource Matching

With the continuous development of the era of big data, different types of sensors, mobile devices, edge servers, and fog nodes will be connected to the core network, and the number of devices connected together will be in the hundreds of millions. In the face of so many resource-rich devices and edge nodes to choose from, we should not directly take all the resource nodes found as input before making an offload decision, which will increase the complexity of the offload optimization algorithm and make it difficult to converge. As with a complex neural network model [144], it is better to preprocess the collected raw data first, rather than directly using the collected data as input to the neural network. Resource matching plays the role of "data preprocessing".

For two different computing paradigms, the first thing we have to do is to identify malicious nodes [145], exclude malicious nodes by judging the data integrity of the nodes found, and then because the user needs of the two computing paradigms are not only computing, but also storage, acceleration networks, etc., so in addition to the initial matching and screening of resources such as computing, communication, energy and other resources on nodes [146], it is also necessary to filter out devices with insufficient processing power or insufficient energy, thereby reducing the dimensionality of input data for offloading decisions. Then, we also need to match and screen these nodes for reliability, security, social ratings, etc.

3.6.4. Task Offloading and Resource Allocation

Task offloading is the transfer of resource-intensive computational tasks to an external, resource-rich platform. Partial or full task offloading is usually done to accelerate resource-intensive and latency-sensitive applications [147]. Resource allocation is usually directly associated with task offloading, for the edge computing paradigm, usually we not only have to give offload decisions (Binary offloading [148, 149] or partial offloading [150, 151]), but also under the response time, energy constraints, or other constraints, give the resource allocation scheme of all devices [152, 153, 154], in order to meet the needs of users with different preferences.

3.6.5. Load Balancing

For task offloading, we generally formulate the offload strategy from the user's point of view, in order to respond to the user's needs faster and reduce the energy consumption of the user's device [155]. However, the reality is that there may be many users who choose the same edge server or fog node for task offload in a period of time. Due to the resource heterogeneity of each device node, in the case of many task requests, there may be some resource-rich nodes with a too-heavy load, and some nodes will have a too-light load, then there will be a waste of resources for the entire system, and may lead to many user processes waiting for too long a time [156]. Then, in order to improve resource utilization more effectively and speed up the response, we must fully consider the load of the system when making the offload decision, transfer the task data from each user device to all edge servers or fog nodes equally, or optimize the processing sequence of the task data of each user [157], which can not only alleviate the waste of resources, but also shorten the waiting time of many processes and achieve load balancing of the entire system [156, 158]. The load we generally consider can be CPU load, amount of memory used, latency, or network load. Load balancing is defined as a technique that divides workloads into multiple devices (such as edge servers or fog nodes), so load balancing not only considers the needs of users, but also improves the resource utilization of the entire system from the perspective of the system.

3.6.6. Resource Orchestration

A lot of research work is to take load balancing into account before making offload decisions, but the reality may be that after the offload decision, some nodes are still selected by many user devices at the same time, resulting in high latency and low bandwidth of the entire system, for example, when some edge servers or fog nodes have a good

signal-to-noise ratio, or contain a lot of popular cache content and high processing power [159]. These servers or nodes are often used by a large number of user devices. Therefore, we need to perform resource orchestration of task offloading between nodes [160] to improve their service capabilities and the load balance of the entire system. Resource orchestration refers to the coordination of resource allocation of the entire system by migrating offloaded user task data, etc., to each node.

3.6.7. Application Placement

In addition to the user's data needing to be offloaded, sometimes we also need to place the application or model on the user's device on the edge server or fog node, such as some latency-sensitive IoT applications: interactive online games, face recognition, etc. Application placement [161] means that all or some of the compute-intensive components of an IoT application (e.g., services, modules, applications, or models) can be placed (i.e., offloaded) executed and stored on edge servers or fog nodes to reduce the execution time of IoT applications and the energy consumption of IoT devices.

3.6.8. Resource/Server Consolidation

In order to ensure that the placement of applications and the offloading of computing tasks can improve the performance of the entire system on the basis of completing user needs, we can not only re-orchestrate the user's computing tasks or applications, but also consolidate the resources of the entire system, such as server consolidation with the help of Virtual Machine (VM) migration technology [162]; Save more energy for the entire system at the cost of increasing the latency of single or multiple users [149]; Or from the perspective of resource utilization [163], when the resource utilization is reduced to a certain threshold, resource migration is carried out to achieve the purpose of consolidating resources.

3.7. ML-based Resource Management in Fog/Edge Computing

In general, ML algorithms can be broadly classified into (i) supervised learning and (ii) unsupervised learning. Supervised learning aims to develop a model from a collection of training instances $((X_1, Y_1), (X_2, Y_2), \ldots (X_i, Y_i))$ where X_i and Y_i represents the predictor and label respectively. In unsupervised learning, the algorithms discover hidden patterns and learn the structure of the training data. In the context of offline learning, the models learn over all the observations in a dataset at a go. First, we discuss the problems related to resource management followed by different AI/ML-based offline learning techniques.

3.7.1. Supervised Learning

It works by predicting Y outputs using the X inputs given to the algorithm to learn from [164]. In short, it consists of ML methods that generate functions with training data. It is generally classified in two ways classification and regression algorithms [165]. It gives better results in complex problems than unsupervised ML algorithms. On the other hand, since the prediction results depend on the training data, the prediction success rate will decrease when there is bad training data. Intelligent Offloading problems in Edge computing can be solved using Supervised Learning [166].

3.7.2. Unsupervised Learning

Contrary to supervised learning, is the learning of correlations in data without input-output tags between data [167]. In short, they are ML methods that produce functions according to the densities of the data and their neighborhood relations. It consists of two main approaches: Dimensionality reduction and Cluster analysis [168]. Dimensionality reduction is also converted to a low dimensional space [169], as high data will require more processing load. Cluster analysis involves grouping clusters of objects with higher correlations to each other [170]. Resource management in Edge and Fog computing using Unsupervised Learning is still an open research area.

- **K-means Clustering:** K-means clustering refers to the method of vector quantization to partition m observations into k clusters where each observation belongs to the cluster with the nearest mean (cluster centers or cluster centroid). K-means clustering is one of the popular methods in resource allocation that can be used for clustering different types of devices based on the available resources in fog/edge computing environments. Such resources can be allocated according to the QoS requirements of each cluster.

3.7.3. Semi-supervised Learning

It is a machine learning method used to combine lowly labeled data with high rates of unlabeled data. It is generally used where Natural Language Process (NLP) is used. It is frequently used for computation offloading problems in Edge and Fog computing [171].

- **Graph Neural Network (GNN):** GNN analyzes data represented as graphs for extracting inferences on node-level and edge-level. Graph theory can be adopted where the network can be represented as graph topology. Chen *et al.* [172] propose a GNN-based framework for resource allocation in wireless IoT networks. The framework specifically deals with the computational and time complexity for conventional resource allocation and outperforms two tasks, namely, link scheduling in Device-to-Device (D2D) networks and joint channel and power allocation. Wang *et al.* [173] present aggregation graph neural network for resource allocation in decentralized wireless networks.

3.7.4. Reinforcement Learning

Standard RL is based on an agent being in connection with the environment by way of perception and action. The agent performs an action based on the environment. The RL model is trained in an iterative manner. The agent upon receiving an input (I) along with an indication of the current state of the environment (S), the agent then chooses apt action, which is triggered as output [174]. The agent works with the objective to maximize reward points. The state can be defined as the snapshot of the environment at that instant particular in time. In the past decade, RL has expeditiously drawn interest amongst the machine learning and artificial intelligence communities. It is one of the dominant and potential techniques, which is immensely being utilized in several domains, including industry and manufacturing. Q-Learning and State-Action-Reward-State-Action (SARSA) are prominent algorithms of this category that have been widely preferred by researchers in the arena of Fog/Edge computing [175]. Fog nodes often face challenges in context to mobility amongst VMs/Containers and location awareness. Concurrently, it becomes expensive to move the VMs to a new location for which RL provides an efficient solution [176].

Resource Allocation Strategies: RL components such as action space, state space, reward, and Markov Decision Process (MDP) emphasize decisions in different computing paradigms. The algorithms for predicting and deciding which resource to be allocated and when, i.e. optimized resource allocation can be done by following algorithms:

- *Deep Neural Network (DNN)*: A DNN is a category of an Artificial Neural Network (ANN) with many hidden layers placed between the Output and Input layers [177]. It can perform real-time allocation of resources as it requires a simple operation. The decisions are made based on experiences and learning, it is different from neural networks in terms of creativity and complications and, hence, gives a global solution with minimal input data [178].

- *Deep Q-Learning (DQN)*: Deep Q-Learning is a simple form of RL that utilizes action or Q-values that enhance the behavior of a learning agent iteratively. In Deep Q-learning, the initial state is input to the neural network which in return output all possible Q-values. It was developed by DeepMind in 2015 giving the benefits of both reinforcement learning and deep neural networks [179].

- *Double Deep Q-Network (DDQN)*: Double DQN uses Double Q-Learning to minimize overestimation by breaking down the max operation in the target to action selection and evaluation. The difference between DDQN and DQN is that DDQN uses the main value network for selecting an action [180].

- *Deep Reinforcement Learning (DRL)*: It is a sub-field of ML that combines the benefits of both Deep Learning and Reinforcement Learning. It is able to input large data sets and predicts what action to perform for optimizing an action. The two sub-algorithms are used in this paradigm, namely, model-based and model-free reinforcement learning algorithms [181].

Table 4 shows the summary of RL-based resource allocation strategies for Fog/Edge Computing.

3.8. Performance Metrics

Optimisation and comparison of any AI-based fog and edge computing architectures are done on the basis of one or more performance metrics, hence these metrics play an important role in the analysis of this architecture and also help to define the merits and demerits of an architecture [187]. Performance metrics are mostly dependent on the type

Table 4
RL-based Resource Allocation Strategies for Fog Computing (FC) and Edge Computing (EC)

Work	Research Focus/Application Area	Paradigm	Method/ Algorithm	Parameter	Result
[176]	Task offloading energy efficiently in Vehicular Fog Computing (VFC) for smart villages	FC	Fuzzy Reinforcement Learning, Integrated on-policy reinforcement learning technique (SARSA) and Greedy heuristic	Total task service time, energy consumption, and average response time	Outperforms over other algorithms up to 15.38% and 46.73% in terms of query response time and energy consumption respectively.
[175]	Computation offloading in Virtual Edge computing systems (Sliced Radio Access Networks)	EC	Integrated Double Deep Q-Network with Q-Function decomposition technique (online Deep-state-action-reward-state-action-based RL algorithm (Deep-SARL))	Maximizing Long term utility performance	Outperforms over three baseline schemes, namely, mobile execution, server execution, and greedy execution
[182]	Intelligent offloading system for vehicular networks	EC	Mobility-Aware Double DQN (MADD), Dynamic V2I Matching Algorithm	Task scheduling and resource allocation (Quality of Experience)	Proposed MADD algorithm performance is 20% and 12% higher than greedy and DQN method, respectively
[183]	Green Fog Computing (Battery management)	FC	Markov-Based analytical model integrated with reinforcement learning process	Job Loss Probability	Effect of Battery Energy Storage System (BESS) varies on the system according to the number of servers
[184]	Resource allocation edge computing network for multiple user	EC	Deep Q-Learning	Data packet size, Channel quality, and waiting time	Deep Q-learning outperforms the random and equal scheduling
[185]	Intelligent Resource Allocation Framework (iRAF) for Edge paradigm	EC	Deep Neural Network for prediction and Monte Carlo Tree Search (MCTS) approach for generating training data	Network states and task characteristics like utilization of edge network resources, the channel quality, latency requirement of services, etc	iRAF achieves 51.71% and 59.27% performance over deep learning and greedy search methods respectively
[181]	Task offloading scheme on priority basis for vehicular Fog Computing	FC	Soft Actor-Critic (SAC) based Deep reinforcement learning algorithm	Entropy of policy and Expected utility to be maximized	High priority task completed preferentially while having better performance of task completion and ratio offloading delay
[186]	Resource allocation in Internet-of-Things network	EC	ϵ-greedy Q-learning	Long term weighted sum cost (task execution latency and power consumption)	Achieved a better trade-off between edge and local computing modes

of layers/ computing model where the architecture is performed. In general, any Fog/Edge computing architecture can be separated on the basis of the following 4 layers.

- **IoT Layer**: This layer is regarded as the first layer of any architecture. This layer is defined as where the IoT devices like Raspberry pi or Arduino can do computation and can coordinate with other sensor nodes and forms a mesh topology-based network. In this layer, the devices are responsible for sensing the data from the sensors and doing some minor operations. This layer can be implemented without any interaction with edge, fog or cloud layers.
- **Edge Computing Layer**: This layer comes next to the IoT layer. This layer consists of switches and routers which are generally termed gateways. This layer acts as an entry point to the fog and cloud layers. It is responsible for

workload distribution and traffic monitoring. It is also responsible for a few less expensive operations resulting in minimizing the response time and optimizing the latency.

- **Fog Computing Layer**: Fog can be defined as the combination of edge and cloud. This layer is located near to edge and IoT layer, and has the capability to perform the expensive operation in comparison to the edge layer. This layer helps to respond to the request faster by computing the work rather than sending the request to the cloud.
- **Cloud Computing Layer**: This layer is the ultimate layer and most powerful layer. The operations which are highly expensive and cannot be performed by any of the previous layers are performed in this layer.

The performance of the above-mentioned layers is measured in multiple terms.

3.8.1. Monitoring Related metrics

These metrics are responsible for monitoring the performance of the entire architecture. Few such metrics are

1. ***Resource Utilisation***: This metric is defined as the amount or the percentage of the resource used or occupied by a specific incoming workload.
2. ***Throughput***: It can be considered as a ratio of the number of tasks arrived at to the number of tasks processed for a certain interval of time.
3. ***Resource Load***: It is defined as the measure of the number of tasks waiting in the queue to be executed along with the number of tasks running.
4. ***Latency***: It is the amount of time gap between actual response time and desired response time.
5. ***Maximum Running Resource***: It is the highest number of resources used.
6. ***Virtual Machine Runtime***: It is the time for which the VM is borrowed.
7. ***SLA Violation***: It is defined as the number of tasks that have been delayed more than the time conceded.
8. ***Energy Consumption***: It is described as the measurement of the energy required by a source to finish the execution of a certain workload.
9. ***Fault tolerance***: It can be defined as a ratio of the number of faults detected to the number of faults that exist.

3.8.2. Analysis related Metrics

Analysis-related metrics are used for the analysis of the performance using monitoring-related metrics. Statistical methods like machine learning or deep learning can be used for this purpose.

1. ***Statistical Analysis***: This is the process where a huge amount of time series data is statistically analyzed and some meaningful information is extracted.

3.8.3. Planing Related Metrics

Planning is the phase in which decision regarding optimization is taken such as VM migration and VM placement.

1. ***Decision Number***: It is referred to the total number of decisions taken.
2. ***Contradictory Decision***: It is the number of times an already made decision is reversed, due to an incorrect decision.
3. ***Completion Ratio***: It is referred to as the ratio in which sources compete for resources.
4. ***Cache Hit Ratio***: It is referred to as the success of a service caching system in reducing the data transmission across the network.

3.9. Simulators

Simulators are the first experimental setup in which architecture is tested before deployment. As any architecture has multiple layers, simulators differ from layer to layer.

3.9.1. IoT layer Simulators

IoT layers initially experiment in this environment. Two Popularly known simulators are

1. ***SysML4IoT***: Abstractions are provided by SysML4IoT to precisely specify various hardware and software services, data flows, and personnel [188].
2. ***IOTSim***: IOTSim [189] is a simulator that uses the MapReduce model, for IoT Big Data processing and simulations in the cloud computing environment. Using this simulator makes the work easier and more cost-effective instead of renting entire large-scale data centers.

3.9.2. Edge Layer Simulators

The edge layer is the next layer to the IoT layer, this layer consist of the Gateways and switches. A few well-known simulators used for the simulation of Edge layers are

1. ***PureEdgeSim***: PureEdgeSim [190] is a large-scale simulation framework for studying the IoT as a distributed, dynamic, and highly heterogeneous infrastructure, as well as the applications that run on these things. It includes realistic infrastructure models, allowing for research on the edge-to-cloud continuum. It covers all aspects of edge computing modeling and simulation. It has a modular design, with each module addressing a different aspect of the simulation. For example, the Datacenters Manager module is concerned with the creation of data centers, servers, and end devices, as well as their heterogeneity. The Location Manager module, on the other hand, handles their geo-distribution and mobility. Similarly, the Network Module is in-charge of allocating bandwidth and data transfer.
2. ***IoTsimEdge***: IoTsimEdge [191] is the extension of IoTsim, which provides the testing of the Edge layer of architecture. This helps its user to test the heterogeneous edge and IoT layer in a configurable manner. It is very user-friendly and easy to use.
3. **SimEdgeIntel**: SimEdgeIntel [192] is an edge simulator that provides the facility to easily deploy mobile with edge intelligence. It provides researchers with detailed configuration options such as customized mobility models, caching algorithms and switching strategies to test their resource management techniques.

3.9.3. Fog Layer Simulator

A few majorly known Fog layer simulators are:

1. ***iFogSim***: iFogSim [193] allows researchers to develop, deploy and test their IoT applications in fog-cloud infrastructure to test custom-made resource management strategies. It provides a hierarchical fog architecture simulation with the first layer made of sensors and actuators for the generation of data, and other layers simulate fog and cloud computing, network, and storage resources.
2. **iFogSim2**: iFogSim2 [194] is an advanced version of iFogSim [193]. It offers advanced features like migration, mobility support, dynamic distribution, and microservice orchestration with resource management.
3. **RelIoT**: RelIoT [195] is NS-3 simulator-based reliability framework for IoT networks. It enables researchers to design customized network reliability management strategies by providing reliability-oriented analysis and predictions early in the design cycle.
4. **YAFS**: YAFS (Yet Another Fog Simulator) is a discrete event-based simulator [196] used to model complex IoT application scenarios in fog infrastructure. With the placement, scheduling, and routing strategy modeling facility, it also provides dynamic module allocation and user movement features.
5. **COSCO**: COSCO [197] is a python-based simulator that provides the facility to develop and test scheduling policies for an edge, fog, and cloud-integrated environment. It provides seamless integration of scheduling policies with a simulated back-end for enhanced decision-making. It also supports real deployment in real-world applications.
6. **DeepFogSim**: It is an extension of VirtFogSim which provides the execution of applications described by generally directed Application Graphs [198]. DeepFogSim simulates the Conditional Neural Networks (CNNs) with early exit on customized fog topology and performance of dynamic joint optimization and tracking of the energy and delay performance of Mobile-Fog-Cloud systems.
7. ***iThermoFog***: This simulator is used to measure the heat or temperature of Cloud Data Centers (CDC) [130]. This simulator uses a Gaussian model to approximate the thermal characteristics of the fog layer server, and optimize the average temperature by scheduling the task.
8. **FogNetSim++**: It is an extension of FogNetSim and provides the facility to simulate both networks with computing modeling aspects of fog computing [199]. It supports low-level network details such as switching and packet routing.

3.9.4. Cloud layer Simulator

Cloud layer is the final layer of any architecture, which is responsible for storage and high-capacity computation. A few major Cloud layer simulators are

1. ***CloudSim***: CloudSim [200] is the most widely used simulator for the simulation of Cloud layer architecture. Many modifications and advances of CloudSim are brought like Dynamic Cloudsim, Container CloudSim, Network CloudSim, and CloudSim Plus. The most recent version of CloudSim is CloudSim5, which combines various releases including containers, VM extensions with performance monitoring features and modeling of Web applications on multi-clouds. This version of CloudSim can also work with other Software-Defined Networking (SDN)/Service Function Chaining (SFC) simulation functions.
2. ***ThermoSim***: ThermoSim [201] is similar to iThermoFog [130], but the CDC whose thermal profile optimization is the cloud layer CDC. This simulator reduces the temperature of cloud CDC by proper scheduling of tasks.
3. **IoTSim**: IoTSim is built on top of Cloudsim simulator, it simulates the processing of IoT big data using the MapReduce model in cloud [189].

3.10. Workloads

Different researchers have used multiple execution traces and benchmarks for the simulation of workloads for AI applications. Some of the well-known workload traces and benchmarks are :

1. ***DeFog***: DeFog consists of five computation-intensive AI applications. These applications cover a diversity of workloads such as deep learning-based object classification applications (YOLO), speech-to-text conversion applications (PocketSphinx), geo-location based online mobile game applications (ipoke-Mon), IoT edge gateway applications (FogLamp), real-time face detection from video streams application (RealFD) and Text audio synchronization or forced alignment (Aeneas). This benchmark captures application-specific system performance metrics for different application domains [124].
2. ***AIOT BENCH***: This benchmark is designed for evaluating IoT device intelligence. It covers different application domains such as image recognition, speech recognition, and natural language processing [202].
3. ***RIoTBench***: It is a real-time suit that captures both system level such as CPU, memory, network, and storage performance metrics, and application-specific system performance metrics for different application domains. It evaluates distributed stream processing systems for streaming IoT applications. It contains 27 IoT tasks classified across multiple categories [203].
4. ***Edge AIBench***: This benchmark model four application scenarios namely: Smart home, autonomous vehicle, Intensive Care Unit (ICU) patient monitoring, and Surveillance Camera for workload collaboration between three layers. Given models can be executed using a federated learning framework available publically [204].
5. ***Bitbrain***: Bitbrain traces consist of performance metrics of more than one thousand hosts of the heterogeneous cloud data center [205]. These traces are categorized into two categories: 1) FastStorage and 2) Rnd workload trace datasets. They consist of models of CPU, RAM, Disk and Bandwidth utilization characteristics. These traces are related to business-critical workloads.
6. ***Azure2016***: This dataset contains VM workload traces captured from November 16, 2016 to February 16, 2017. The information captured in this dataset includes VM id, its subscription, VM role name, cores, memory, and disk allocations, and minimum, average, and maximum VM resource utilization [206].

4. AI Based Techniques For Resource Management in Fog/Edge Computing

In this section, we discuss resource management-focused AI/ML techniques for enabling Edge/Fog AI and present a summary of AI/ML-based resource management techniques for Fog/Edge Computing. Figure 7 shows the taxonomy of AI-Based Techniques For Resource Management in Fog/Edge Computing.

4.1. Machine Learning Techniques for Edge AI Management

Machine learning is showing remarkable results in various fields. The promising results of machine learning in various domains are also attracting researchers' attention to the use of ML for modeling, classification, prediction, and forecasting related to resource management in fog computing for enabling Edge AI [207]. When we discuss multi-tenant environments where infrastructure is used by many applications which have different requirements, it is very obvious that tunning one application may have impacts on other applications [208]. Also adding complexity to the problem, these applications are deployed on multiple heterogeneous distributed resources. When dealing with different workloads, and heterogeneous distributed resources, it is better to analyze the workload from application and infrastructure perspectives. Understanding workload behavior can lessen the complexity of the problem and improve

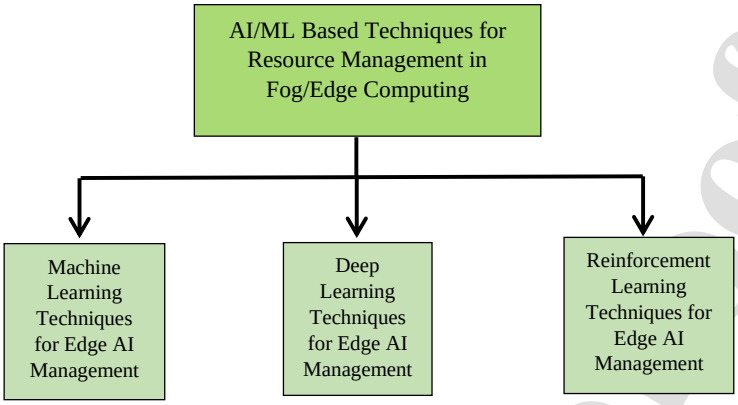


Figure 7: Taxonomy of AI-Based Techniques For Resource Management in Fog/Edge Computing

the performance of an application in edge AI [209]. Different works in the literature considered workload analysis using ML algorithms. The IoT-enabled edge AI is also prone to temporal effects. There can be an increase in workloads on certain resources at certain times. The prediction of workload behavior or spatio-temporal effects in advance can ease the resource orchestration process. To facilitate auto scaling of resources, Liu *et al.* [210] proposed a workload pattern discrimination-based adaptive prediction approach for infrastructure as a service. Due to the speed of workload change, they classified workload into two groups: fast time scale data and slow time scale data. Fast time scale data had the feature of stochasticity and nonlinearity while slow time scale data had the feature of linearity. For two datasets they used Support Vector Machine (SVM) and Linear regression (LR) for the prediction of workload. Another work proposed use of auto-correlation measurement and similarity clustering for CPU workload prediction on VMs [211]. A combination of random forest, SVM and neural network is used to predict future workloads to reduce training time and increase the accuracy of the model [212]. Work in reference [213] addressed the issue of workload management using decision trees. As fog environments are distributed and heterogeneous, and diverse IoT-based AI applications with different resource requirements make a selection of optimal nodes for application placement to satisfy QoS and Quality of Experience (QoE) constraints, more challenging. Addressing the application placement problem in mobile fog, Rahbari *et al.* proposed to use of classification and regression trees. In order to manage power consumption in edge/fog-based smart building services, work in [214] used k-nearest neighbors (KNN) and decision tree algorithms.

In another work, they addressed the application placement problem for smart city applications. They employed logistic regression and support vector machine for job completion time approximation [215]. Addressing the resource scheduling problem, Liu *et al.* [186] combined fuzzy c-mean clustering with particle swarm optimization. Using optimized fuzzy c-mean clustering they tried to reduce the scale of resource search. They compared the proposed work with Fuzzy c-mean clustering and the objective function value of optimized fuzzy c-mean showed faster convergence speed than the Fuzzy c-mean algorithm. Yadav *et al.* [145] also used fuzzy c-mean clustering for task allocation in distributed systems to minimize the cost of the system. To minimize delay for IoT-based applications in fog environment, Shooshtarian *et al.* [216] used hierarchical clustering to find the nearest neighbour node to the IoT device to solve the resource allocation problem. Container orchestration is also investigated using ML methods by many researchers. Researchers in reference [217] used a time series analysis model (ARIMA) combined with the docker container technique for resource utilization prediction in containerized applications. Another work explored performance analysis of containerized applications using polynomial regression and k-means clustering. They classified multi-layer container execution structures based on the application performance requirement in distributed resources [218]. Authors in [219] used SVM, Boosting decision tree, Random forest and Naive Bayes for node performance prediction to improve resource scheduling decisions. Podolskiy *et al.* [220] used Lesso for the self-adaptive problem of vertical elasticity for co-located containerized applications. Figure 8 presents a summary of machine learning algorithms that are used for the analysis and prediction of workload and resource usage to aid resource management in edge/fog computing.

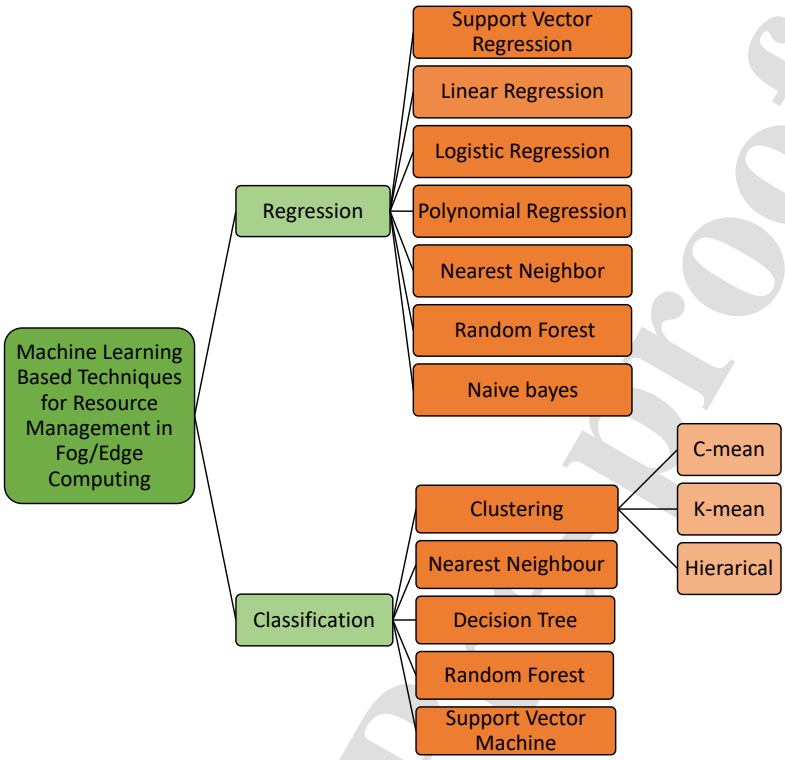


Figure 8: Summary of Machine Learning-based Resource Management for Fog/Edge Computing

4.2. Deep Learning Techniques for Edge AI Management

Currently, deep learning-based prediction models are the most promising architectures for computational intelligence. It shows good performance in various problems such as workload prediction, where traditional machine learning algorithms fail. CNNs can be used to model wide spatial dependencies by extracting local features by adopting layers with convolutional filters [221]. Long Short-Term Memory (LSTM) can be utilized for the prediction of fluctuating and volatile workload time series due to its capability to capture long-term temporal dependencies [222]. For time series analysis, authors in [223] have presented a deep learning model based on the canonical polyadic decomposition for workload prediction for industry informatics. Sima and Saeed [224] used CNN for predicting future cloud workload in advance for optimized resource allocation. For dynamic management of network resources, Bega *et al.* [225] also used CNN in their work. Their proposed strategy returns a cost-aware capacity forecast, which can be directly used by network operators to take re-allocation decisions that maximize their revenues. Authors in reference [185, 226] addressed the resource provisioning issue with Fully Convolutional Networks (FCNs). Tuli *et al.* [22] focused on straggler detection for the system's QoS and used an encoder LSTM network for the analysis of tasks. Their proposed model analyzes the tasks and predicts which tasks can be a straggler. Work in [227] also used Bi-LSTM to address the scheduling issue of fog-enabled Radio Access Networks (F-RAN). For optimal performance, they used Bi-LSTM for the prediction of content popularity. Some of the recent work also used DNN as a surrogate model for QoS prediction to make scheduling decisions [177, 178]. Considering accurate resource requests prediction essential for achieving efficient task scheduling and load balancing, Zhang *et al.* [228] used Deep Brief Network (DBN) for day and hour scale predictions of CPU and memory utilization. They evaluated their proposed technique with the Google dataset. Many works also improved prediction performance by ensembling multiple algorithms for workload prediction for fog computing [229]. To provide real time responses to vehicular applications, such as traffic and accident warnings in

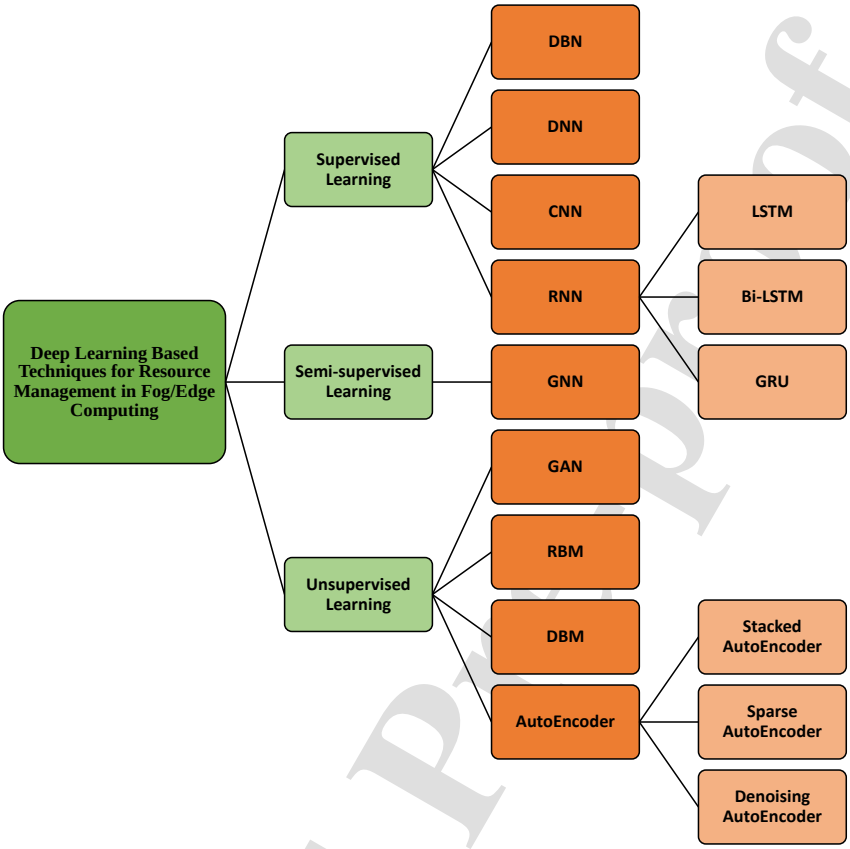


Figure 9: Summary of Deep Learning-based Resource Management for Fog/Edge Computing

the highly dynamic Internet of vehicles (IoV) environment. Lee *et al.* [230] used LSTM-based Deep Neural Networks (DNNs) to predict mobility behaviour and movements of vehicles. They combined RL with LSTM-based DNN for resource allocation in Vehicular Fog Computing (VFC). In their other work, they also used Recurrent Neural Network (RNN) for resource allocation problems. To extract the time and space-based pattern of resource availability, they integrated the RNN into the DNN of the proximal policy optimization algorithm [230]. Another work [231] used a hybrid CNN-LSTM model for the prediction of multivariate workload in an attempt to extract complex features of the VM usage components, then model temporal information of irregular trends in the time series components. They evaluated the proposed model for resource provisioning using bitbrains dataset and compared it with other predictive models. To minimize the complexity and non-linearity of the prediction model, Yazdanian *et al.* [229] decomposed workload time series into its constituent components in different frequency bands and used ensemble Generative Adversarial Networks (GAN)/LSTM for prediction of each sub-band workload time-series. The proposed model employs stacked LSTM blocks as its generator and 1D ConvNets as the discriminator. Graph Neural Network (GNN) analyzes data represented as graphs for extracting inferences on node-level and edge-level. Graph theory can be adopted where the network can be represented as graph topology. Chen *et al.* [172] propose a GNN-based framework for resource allocation in wireless IoT networks. The framework specifically deals with the computational and time complexity for conventional resource allocation and outperforms two tasks, namely, link scheduling in D2D networks and joint channel and power allocation. Wang *et al.* [173] present aggregation graph neural network for resource allocation in decentralized wireless networks. Figure 9 presents a detailed classification of deep learning-based algorithms used for Resource Management in Edge/Fog computing.

4.3. Reinforcement Learning Techniques for Edge AI Management

As resources are heterogeneous and capacity-constrained in edge, smart resource allocation is considered one of the important factors in enabling Edge AI. Considering edge/fog a pool of different resources like CPU, GPU, storage etc. efficient resource allocation necessitates resource sharing. Reinforcement learning algorithms are the most experimented algorithms for resource sharing and allocation decision-making at the moment. Usually, MDP type problems are solved using policy gradient methods, tabular RL and deep Q-learning methods. Tabular methods such as Q and SARSA are not preferred by researchers due to their low scalability for modeling computing systems with thousands of devices.

Shi *et al.* [181] presented a deep reinforcement learning (DRL)-based scheme for task offloading for VFC application. They proposed a soft-actor critic for the maximization of the policy of entropy and anticipated reward. Fu *et al.* [232] utilized a maximum entropy framework-based soft actor-critic DRL algorithm in VFC-enabled IoV for providing low bitrate variance live streaming service for vehicles. To reduce the vehicle's long-term mean cost with promising reliability and latency performance in VFC, a Deep Q Network (DQN) is presented for the switching problem. They designed the mobile network operator (MNO) preference and switching problem by simultaneously analyzing switching cost, cost variation by MNO and fog servers, and QoS variation within MNOs [233]. In another work [234], a dual neural network of Deep Q-Learning method is implemented for resource slicing management. They formulated a semi-MDP for the simultaneous allocation of resources. Considering computational offloading an important factor for enabling edge AI, another work considered energy-efficient vehicle scheduling for task offloading in VFC. To resolve the high dimensionality issue caused by the increased number of vehicles in road-side units (RSU) coverage, an on-policy reinforcement learning-based scheduling algorithm combined with a fuzzy logic-based greedy heuristic, named Fuzzy Reinforcement Learning (FRL) is proposed. This greedy heuristic not only accelerates the learning process, but also improves long-term reward when compared to Q-learning algorithm [176]. Chen *et al.* [175] addressed offloading issue of virtual edge computing. They formulated the offloading problem in a sliced radio access network as MDP. They resort to DNN based function approximator and drive a double deep q network for making offloading decisions. Cheng *et al.* [235] proposed a policy gradient learning-based scheduler for task scheduling in edge devices. The same approach Multi-agent Deep Deterministic Policy Gradient (DDPG)-based scheduling is adapted for joint task partitioning and power control in fog computing networks.

Ning *et al.* [182] explored deep reinforcement learning for optimization of task scheduling and resource allocation in vehicular networks. They divided the problem into two sub-optimization problems. First is deciding the priority of the vehicles for the quality of experience of users using a utility function. The second subproblem of resource allocation is formulated as the DRL problem. A deep Q network is improved by applying dropout regularization and double deep Q networks to deal with the defect of overestimation. To address the resource provisioning issue in fog, work in reference [236, 237] used Deep RL based on DQN. In addition, some authors experimented with Policy gradient learning for efficient resource provisioning resources [238]. One other work used A3C (Asynchronous Advantage Actor-Critic) and residual neural network for scheduling stochastic edge-cloud environment [239]. some work also used the same RL model for workflow scheduling [240, 241].

Liu *et al.* [186] also addressed the resource allocation problem for IoT-enabled edge AI and proposed a $\epsilon-$ greedy Q-learning-based optimum offloading algorithm. The problem is formulated as a weighted sum cost minimization problem with its objective function including the task execution latency and the power consumption of both the edge device and the end device.

Since in the majority of IoT-based Edge AI systems, sensors produce a lot of data that needs to be processed within the deadline of applications, the inherent lack of information in tasks arrival of such systems necessitates adaptive task scheduling. Intelligent task scheduling not only minimizes task execution delays but also improves system key performance indicators (KPI) like reduced energy consumption, and load balancing. DRL methods have shown some promising results in decision-making problems. There are several works in the literature that explored RL for adaptive task scheduling. To minimize computation costs and long-term service delays, a Double deep Q learning (DDQL) is proposed in work [180]. In order to achieve optimal action selection, each agent used two separate models for action selection and Q-value calculation. Each RL agent was embedded in the gateway device to schedule tasks and allocate resources to tasks. The RL agent tries to maximize cumulative reward to achieve reduced end-to-end delay. To cease fluctuation in the results, they integrated the target network and experience replay mechanism in the DDQL-based scheduling policy. In order to maximize the long-term value of QoE, Sheng *et al.* [242] designed an intelligent task scheduling system using a model-free DRL algorithm. They formulated a task scheduling problem on heterogeneous virtual machines as an MDP problem and solved it with policy-based DRL. This work considered

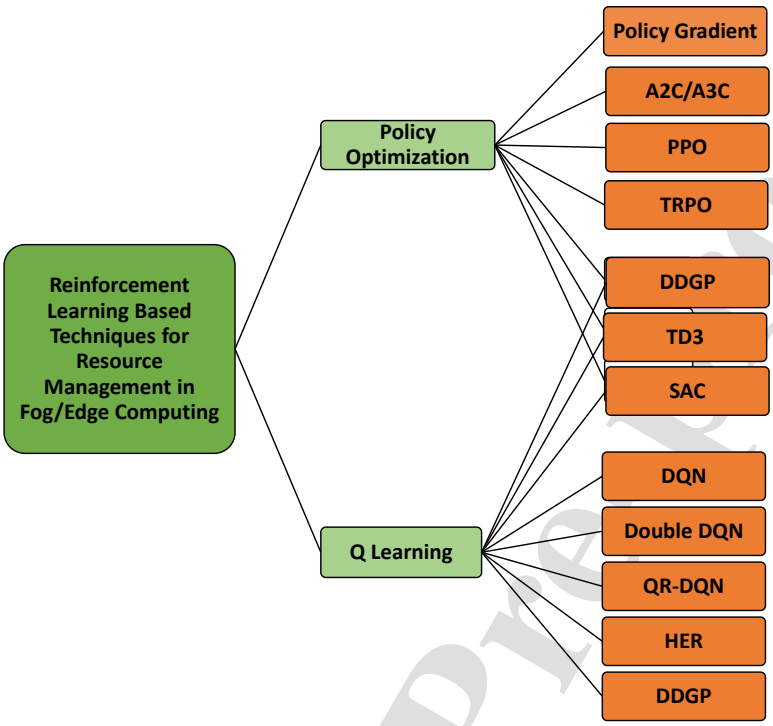


Figure 10: Summary of Reinforcement Learning-based Resource Management for Fog/Edge Computing

task satisfaction degree as reward and action is represented as a pair of tasks and VM. They decoupled real-time steps from scheduling steps in MDP formulation to make action space linear with a product of the number of virtual VMs and queue size and to schedule multiple tasks in a single time step. In order to achieve ultra-low latency and fairness in resource sharing, Bian *et al.* [243] proposed FairTS that ensures fairness between tasks and with ultra-low average task latency. One other factor that can degrade the performance of Edge AI applications is an imbalance in workload distribution between resources in the system. The solution to this issue is offloading or redistribution of tasks. RL is investigated for offloading decision-making in [244]. They formulated Offloading problem as MDP and proposed a DRL-based scheme to make users enable to make near-optimal decisions by considering uncertainties in the user device and cloudlet movements and resource availabilities. Another work used Deep Q Network (DQN) for making optimal actions on how main tasks will be offloaded and how many processed locally [245]. Chen *et al.* [121] propose a two-timescale federated deep reinforcement learning based on Deep Deterministic Policy Gradient (DDPG) to solve the joint optimization problem of task offloading and resource allocation to minimize the energy consumption of all IoT devices subject to delay threshold and limited resources. The simulation results show that the proposed algorithm can greatly reduce the energy consumption of all IoT devices. Lee and Lee [230] utilized proximal policy optimization (PPO) RL for offloading problems in order to provide real-time responses for vehicular applications. PPO with the ability to continuously learn dynamic environments can easily adjust to make resource allocation decisions accordingly. Some works [84, 238] used Policy gradient learning for the deployment of DNN in Edge AI. For efficient real-time resource allocation and offloading in internet of vehicles, Hazarika *et al.* [246] utilized DDPG and twin delayed DDPG (TD3) algorithms. They compared the proposed technique Soft Actor Critic (SAC) and DDPG. Another work formulated resource allocation in Mobile Edge Computing (MEC) as an MDP problem in order to minimize system delay and solved it with hindsight experience replay (HER) improved DQN [247]. Figure

10 presents a summary of reinforcement learning-based algorithms that are used separately or in a hybrid fashion with DL for resource management in Edge/Fog computing.

5. Taxonomy

This section discusses the proposed taxonomy of frameworks and comparison analysis in AI-based edge and fog computing

5.1. Taxonomy of AI-based Fog and Edge Computing

In this section, a comprehensive taxonomy of AI-based fog and edge computing approaches is proposed based on the existing studies following a systematic review. The taxonomy of the framework is shown in Fig. 11, which includes infrastructure that supports the platform, objectives that the proposed approach aims to achieve, deployed platform, the mechanism for resource management, metrics for performance evaluations, the category of AI-based methods, and target application. Each taxonomy is further classified into the detailed study of AI-based fog and edge computing framework.

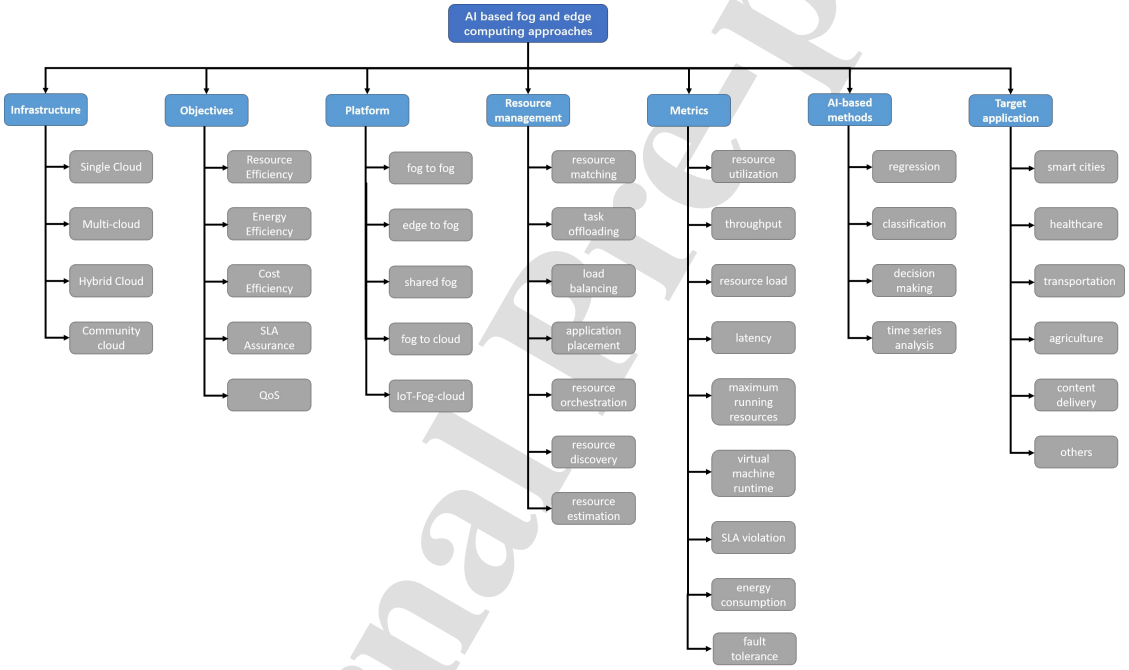


Figure 11: Taxonomy of AI-based Fog and Edge Computing Frameworks

Infrastructure: the AI-based fog and edge computing approaches can be supported by different infrastructure models including single cloud, multi-cloud, hybrid cloud and community cloud with different focuses. For example, multiple clouds can work collaboratively to complete the partitioned deep learning tasks in fog to edge environment.

Objectives: based on our investigation, we notice that the existing major optimization objectives in AI-based fog and edge computing approaches are improving resource efficiency, reducing energy consumption, decreasing cost efficiency, assuring SLA and ensuring QoS.

Platform: the platform indicates how the fog and edge computing approach is deployed. The current mainstream platforms include fog-to-fog, edge-to-fog, shared fog, fog-to-cloud and IoT-Fog-cloud.

Resource Management: one of the key challenges in fog and edge computing environments is managing the resources efficiently. The current research has been conducted for resource matching that maps the suitable amount of resources to tasks, task offloading that processes task collaboratively among fog and edge, load balancing that balances the workloads for different nodes, application placement that deploys the fog/edge applications to devices, resource

Table 5
Comparison of Existing Studies based on Taxonomy

Approach	Infrastructure	Objectives	Platform	Resource management	Metrics	AI-based methods	Target application
Rafique *et al.* [249]	Multi-cloud	Resource Efficiency	edge to fog	load balancing	resource utilization	None	others
Golec *et al.* [97]	Single Cloud	Others (safety)	IoT-Fog-cloud	resource matching	latency	None	others
Golec *et al.* [100]	Hybrid Cloud	Resource Efficiency	IoT-Fog-cloud	resource orchestration	running resources	classification	healthcare
Iftikhar *et al.* [102]	Multi-cloud	Resource Efficiency	edge to fog	resource orchestration	resource utilization	decision making	others
McChesney *et al.* [124]	Hybrid Cloud	Resource Efficiency	edge to fog	application placement	latency	None	others
Aazam *et al.* [136]	Multi-cloud	Cost Efficiency	edge to fog	resource estimation	energy consumption	decision making	others
Aazam *et al.* [137]	Hybrid Cloud	QoS; Cost Efficiency	IoT-Dog-cloud	resource estimation	resource utilization	decision making	others
Ahmed *et al.* [250]	Multi-Cloud	Resource Efficiency	fog to cloud	resource orchestration	throughput	classification	others
Bi *et al.* [149]	Single Cloud	Resource Efficiency	IoT-Dog-cloud	task offloading	resource load	regression	others
Sim *et al.* [143]	Multi-cloud	Resource Efficiency	fog to fog	resource orchestration	None	None	others
Vu *et al.* [156]	Hybrid Cloud	Resource Efficiency	fog to cloud	resource orchestration	resource utilization	None	smart cities
Yadav *et al.* [145]	Multi-cloud	QoS	fog to fog	application placement	running resources	decision making	others
Yao *et al.* [154]	Multi-cloud	QoS	edge to fog	task offloading	latency	decision making	others
Wu *et al.* [251]	Hybrid Cloud	QoS	edge to fog	resource orchestration	SLA violation	decision making	others
Xue *et al.* [252]	Hybrid Cloud	Energy Efficiency	edge to fog	task offloading	energy consumption	decision making	others
Wu *et al.* [248]	Hybrid Cloud	Energy Efficiency	edge to fog	task offloading	energy consumption	decision making	others
Liu *et al.* [210]	Single Cloud	Resource Efficiency	edge to Cloud	Resource Provisioning	Cost, Resource Utilization	decision making	others

orchestration that automates the resource allocation, resource discovery that provides the naming services of available services and resource estimation that predicts the amount the required resources.

Metrics: multiple metrics have been utilized to evaluate the performance of proposed approaches. The dominant metrics include resource utilization, throughput, resource load, latency, maximum running resources, virtual machine runtime, SLA violations, energy consumption and fault tolerance.

AI-based methods: the fog and edge environment has adopted AI-based methods to assist their management. Some existing categories of AI-based methods include regression, classification, decision making and time series analysis, which can be applied to workloads prediction, application feature analysis and making resource scheduling policies.

Target applications: the investigated approaches have been applied to support IoT applications in different areas including smart cities, healthcare, transportation, agriculture, content delivery and etc.

5.2. Comparison of Existing Studies based on Taxonomy

Table 5 summarizes and compares the selected studies of AI-based fog and edge computing frameworks discussed in previous sections in terms of infrastructure, objectives, platform, resource management, metrics, AI-based methods and target application. For example, Golec *et al.* [100] applied their approach in a multi-cloud environment and aimed to improve resource efficiency under the IoT-Fog-Cloud paradigm. They also utilized an AI-based approach for classification in resource orchestration to improve resource utilization. Wu *et al.* [248] exploited the task offloading technique to reduce energy consumption and the proposed approach-based AI can help to make optimized decisions on when and how to manage offloaded tasks. Vu *et al.* [156] considered their scenario for smart cities with a hybrid cloud model to improve resource utilization under fog to cloud environment. Based on our investigation and comparison, we can notice that AI-based approaches have been comprehensively applied in fog and edge environments and more applications can be further incorporated into this paradigm.

To summarize, the existing research works have covered all the types of dominant infrastructures including single cloud, multi-cloud and hybrid cloud. In terms of optimization objectives, most of the works focus on improving resource efficiency and QoS. As for the deployed platforms, Edge-to-Fog, IoT-Fog-Cloud, and Fog-to-Fog are the mainstream ones to support applications. The techniques applied to optimize resource management are diverse, including load balancing, resource matching, resource orchestration, application placement and task offloading. There are several metrics have been widely utilized to measure the performance of the proposed approach from different perspectives, including resource utilization, latency, energy consumption, throughput and SLA violation. The AI-based methods have been exploited for two main objectives, including classification, prediction and decision-making.

6. Result Outcomes

Our study highlights the systematic review of various articles in order to understand the prevailing status of fog/edge computing. The extensive study comprises various driving forces responsible for making an impact on emerging paradigms in the form of open issues and future work. In total, we collected 320 articles, out of which 135 were shortlisted after the iterative selection process. The articles emphasize the state-of-art work done in the domain of research management in fog/edge and how the implication of intelligent paradigms like Artificial Intelligence, Machine

Learning is inciting researchers. The taxonomy of our study has been designed with references to articles from the year 2015 to 2022. As depicted in Fig. 12, the majority of our referred papers are from the year 2022. This accentuates the fact that our survey includes the latest work done by the research community.

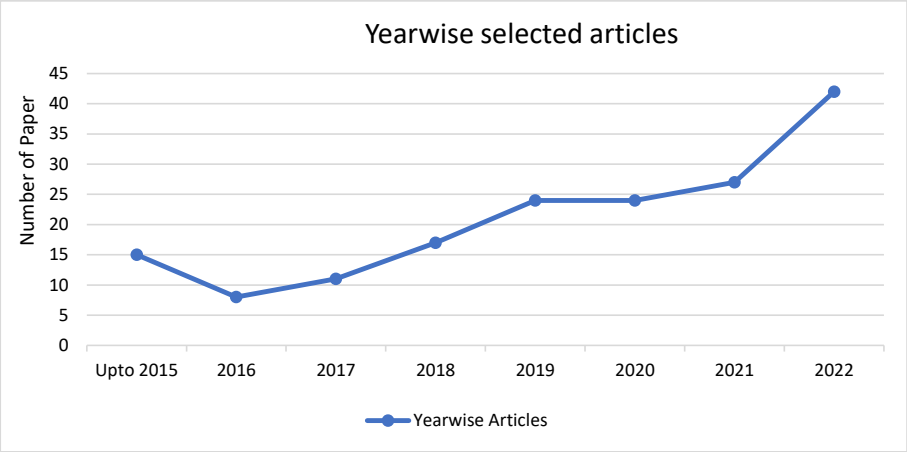


Figure 12: Year-wise Publications of AI/ML based Fog/Edge computing papers

The structure and methodology of the survey are inspired by the Systematic literature review (SLR) procedure by Kitchenham [33]. Furthermore, the identification of research questions channelizes the process flow of reviewing methodology. In a research review, the search process comprises the research topic which plays a significant role. The content in this paper has been accumulated from various sources including ACM Digital Library, IEEE Xplore, Springer Link, and other resources like Scopus, National Digital Library and electronic scientific research databases. Figure 13 describes the yearly bifurcation of various sources in the form of paper count from different publications that represent most of the articles are from IEEE journals, transactions and conferences as compared with other publishers. Further, we have rigorously reviewed every article and divided it into five sections review, survey literature review, Implementation in real and simulation environments and book chapters as shown in Figure 14.

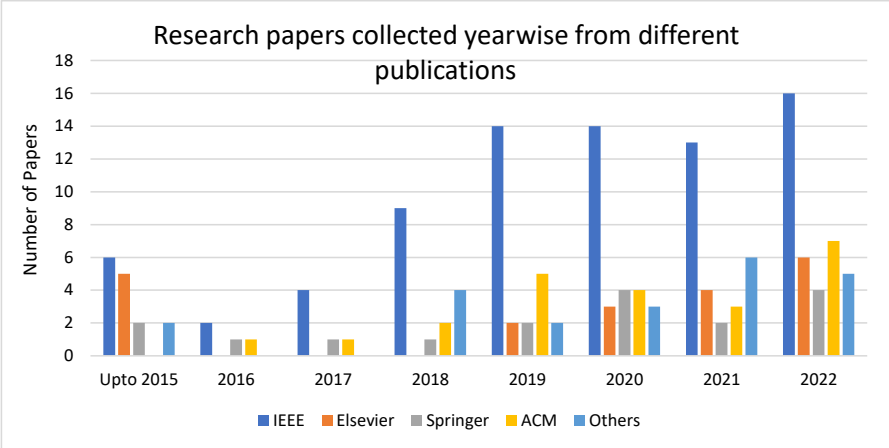


Figure 13: Bifurcation of research papers on the basis of publishers

This study considers various aspects of fog/edge computing which have been categorized into resource relating aspects of resource management further categorized as resource provisioning, task offloading and resource allocation), QoS parameters, and concerning other factors relating to real-world challenges like IoT, healthcare, security and privacy as shown in Fig. 15. A major chunk of our survey is inspired by the resource aspect comprising 61 papers and QoS

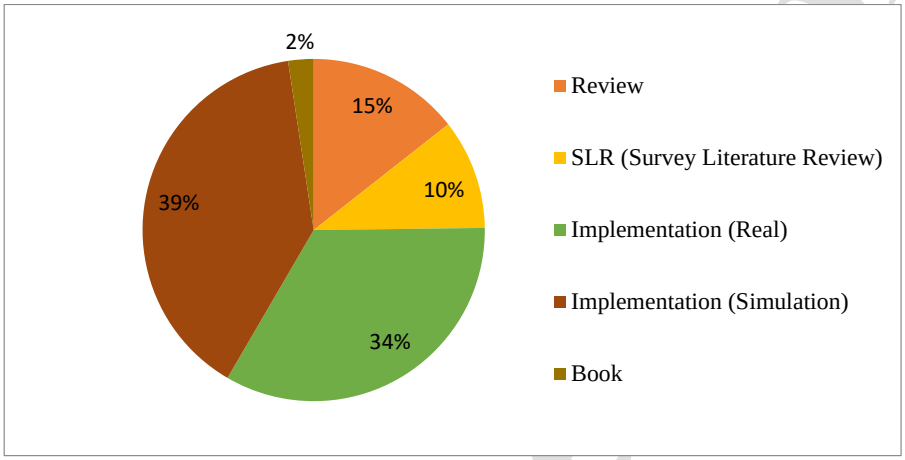


Figure 14: Study type of research paper

parameters including 49 papers. The studies have been demonstrated in chronological order similar to other studies for the identification of state-of-art work in an effective manner.

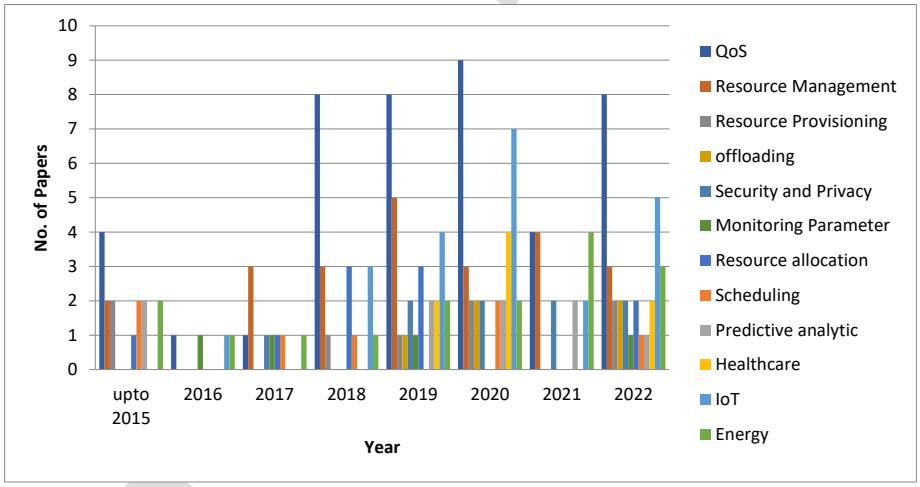


Figure 15: Categorization of papers based on factors relating to fog/edge computing

7. Open Issues and Future Directions

Despite the fact that a significant amount of progress has been made in AI and ML thanks to fog and edge computing. Despite this, there are a great number of problems and obstacles in this area that need to be solved. We have compiled a list of outstanding challenges in this field based on the existing literature.

7.1. Heterogeneity

Fog and edge computing are meant to support IoT applications that will emerge in different programming languages (e.g., C, Python, and Java), hardware architecture (e.g., ARM and AMD), processing units (e.g., CPU, GPU, and TPU), etc. Such heterogeneity appears a highly challenging concern for the problem of resource management. That is, a resource manager, e.g., a resource balancer, requires understanding such differences in decision-making. Otherwise, traditional homogeneous-based solutions would result in considerable resource wastage and inappropriate decisions made by the resource manager. For instance, if some edge devices are ARM-based and others are AMD-based architecture, the resource manager must adhere to this difference since IoT applications may be incompatible with counterpart architectures. Another example is when fog devices provide unequal computation capacities. In this case, the resource manager requires an understanding of the devices' capacity to treat them proportionally. Otherwise, resources will be wasted and the QoS will degrade. When AI-based solutions are approached, the heterogeneity becomes even further challenging since AI models are agnostic to the heterogeneity while they perform completely differently on different devices. For instance, if an edge device to run AI models is enabled with accelerators such as GPU which can provide higher precision and shorter latency, the resource manager requires an understanding of such differences to other edge devices that lack accelerators. Otherwise, costly accelerators will be undermined. Hence, heterogeneous resources at the edge will bring several opportunities for AI-based resource management, only if the traditional solutions are redesigned to support them.

7.2. Environmental Sustainability

Sustainability appears to be a first-class requirement of IoT applications, since many of them rely on renewable energy sources such as solar irradiation. Sustainability is further important since edge devices are presumably constrained in computation resources. With such characteristics, edge devices strive for resource efficiency, in terms of energy or computations. However, this appears to conflict with the resource-hungry nature of AI models that demand a considerable amount of resources to be able to perform as expected.

With the sustainability requirements of edge and resource-hungry features of AI models, it is very challenging to welcome AI at the edge. The hardware sector at the edge side and the software developing sector on the AI have to progress towards this ambition. However, as far as AI-based resource management is concerned, certain considerations can be assessed by the community. That is, a trade off between how much benefit the AI-based solution, as compared to a non-AI-based solution, can achieve and how much resource is consumed is a key question.

Another direction is, instead of asking about using AI-based solutions or not, what sort of AI-based solutions in terms of precision should be used? For instance, models can train to perform on lightweight frameworks to remain edge-friendly, but provide weaker precision. For instance, a resource manager can perform on the TensorFlow framework to make precise decisions, but consume a considerable amount of resources; or can perform on TensorFlow Lite to consume fewer resources upon inferences, but provide weaker precision. Hence, the open question is when and how to utilize AI-based solutions to satisfy the requirements of IoT applications, while achieving desirable precision.

7.3. Security versus Efficiency

AI-based resource management solutions for IoT applications require to adhere to security concerns. While edge computing is to not necessarily rely on the cloud, it brings its own limitations such as security. IoT applications in many domains such as Smart Home or Smart Healthcare, require the edge platform to adhere to such. The resource manager is a key component of an edge platform, hence, requires re-architect to satisfy that. This appears challenging due to the distributed nature of edge platforms. However, when AI-based solutions are enabled, this becomes even further challenging since AI-based models continuously require observations of the IoT applications to obtain sufficient data for the training or inference. Hence, the open issue here is how to satisfy AI requirements without degrading security. From another perspective, the security requirement itself can be driven by AI itself where the resource manager can utilize AI to learn patterns, outliers and features that can affect the security of an AI-based resource manager. An open question here would be how to leave such responsibilities to the resource manager through AI.

7.4. AI for Edge and AI at the Edge

AI-based resource management represents a perfect example of AI for an edge while AI-based applications, such as object detection, classification, etc, represent AI-based applications serving at the edge. Throughout this paper, AI for edge was discussed, but the cohabitation of AI applications and AI managers would raise several new opportunities and challenges. A key open issue is how to shift the resource manager's focus from tuning the resources, e.g., by offloading,

scheduling, etc, to tuning the AI application to achieve the adaptation. This is important because AI applications can be tuned to consume different amounts of resources from CPU to accelerators. Even on CPUs, they can perform differently such that they affect the energy and computation resources variably. Hence, with AI both for and at the edge, there are many opportunities that require investigations.

7.5. AI for Serverless Computing

While new AI applications are being used at the edge for certain use cases such as object detection for a smart traffic light, new application deployment and development models are also entering the edge. Serverless computing with its Function-as-a-Service (FaaS) is one of the major technologies in this context. FaaS is an application development model that turns bulky applications into single-purpose execution units, called functions, that are deployed upon event-based invocations and terminated after executions to save cost and resources. However, if long-running AI-based applications and resource managers stem into this area, several questions would raise that require investigation. For instance, would FaaS, whose billing model highly relies on the execution duration, still be cost-efficient? would this ephemerality (executing and terminating) be a bonus for the AI-based resource manager to remain resource efficient? Would the cold start of a function, the time duration from invoking to launching, deteriorate for AI-based applicants that require loading presumable heavy run-times?

7.6. Resource Federation

Distributed edge and fog devices require shared data and computation to function properly. Using AI-based managers for inference requires such data to be collected regularly and may be of a bigger size than typical raw data. For instance, if a traditional manager collects CPU usage of devices, an AI-based solution may collect other forms of data such as objects. Using AI-based managers for training also requires much more data. Add to this demand the scatteredness and scale of a network in a distributed edge that can span up to thousands of devices. Given a such scale, research areas around decentralized and federated AI-based resource management appear highly important. The decentralization means each resource manager is in charge of a portion of the cluster. The federation means while each portion works in isolation, they can collaborate with peers to use resources or to achieve a collective goal. Such areas have already commenced in edge computing and AI, but little effort has been made, particularly in the area of resource management which requires consideration.

8. Summary and Conclusions

In this work, we have conducted a systematic literature review on how machine learning and artificial learning-based solution are utilized for the resource management problem in fog and edge computing environments. Recent research works have witnessed a quickly growing trend of adopting AI-based methods to address the limitations of traditional heuristic approaches without sufficient consideration of diverse and dynamic factors in the environment. Compared with most traditional heuristic methods, AI-based approaches can be used to make accurate resource management decisions with lower time overhead, model and predict application and infrastructure metrics to improve the quality of services. Our work also advances the relevant surveys by considering fog computing and edge computing together with extensive comparisons

To summarize, we have noticed that AI-based methods have been applied in a wide range of scenarios, including resource estimation, resource discovery, resource matching, task offloading, load balancing, resource orchestration, application placement, and resource consolidation. We also observe that the applications deployed on fog and edge computing environment ranges from healthcare, smart home, agriculture, smart transportation, and spatial. Significant efforts have been made to utilize advanced AI-based approaches, e.g. DNN, Q-learning, DQN, and reinforcement learning-based algorithms, to optimize resource utilization, throughput, SLA violations, energy consumption, and fault tolerance.

In conclusion, although the relevant research progresses fast, there is no systematic literature review that combines fog and edge computing with an AI-based optimization framework in charge of the whole resource management process. Employing microservice and serverless can be a promising approach to further optimize the application and system performance with fine-grained resource control. This taxonomy work will assist the researcher to find the important research directions in edge and fog computing and will also help to choose the most suitable AI-based methods for efficient resource management under the hybrid paradigm under a dynamic environment.

Table 6
Preliminary Examining questions

Question	Yes	No
Q1. Does the article discuss the use of AI/ML in Fog/Edge Computing? This report compiles findings from studies conducted on AI/ML in Fog/Edge Computing. This survey takes into consideration all of the research publications, including case studies, experimental studies, and so on.		
Q2. Is the primary emphasis of this paper the AI/ML-based management of resources in Fog/Edge Computing? Does this paper provide a method, approach, system, or framework for resource management that could be used for AI/ML in Fog/Edge Computing? Is the validity of this investigation ensured by utilising a simulated testbed for fog/edge computing? Is the validity of this investigation ensured by utilising a real testbed for fog/edge computing?		

Table 7
Specific questions

Question	Yes	No
Q1: what resource management methods are available that are based on artificial intelligence and machine learning?		
Q2: Where are AI/ML-based fog/edge computing frameworks stand right now?		
Q3: How can the efficiency of AI/ML-based fog/edge computing be measured, and what metrics are used for this purpose?		
Q4: Which simulators are utilized for fog/edge computing that is based on AI/ML?		
Q5: What are the most common applications of IoT-enabled Edge/Fog AI?		
Q6: What kinds of workloads are utilised to evaluate the efficacy of AI/ML-based fog/edge computing frameworks?		

Acknowledgements

Sundas Iftikhar would like to express her thanks to the Higher Education Commission (HEC) Pakistan (Grant No. 2-5/FDPOS/HRD/UoK/QMUL/2020/1) for their support and funding. This work is partially funded by Chinese Academy of Sciences President's International Fellowship Initiative (Grant No. 2023VTC0006), and Shenzhen Science and Technology Program (Grant No. RCBS20210609104609044).

Conflict of Interest

On behalf of all authors, the corresponding author states that there is no conflict of interest.

Appendix A: A quality assessment forms

A.1. Preliminary Examining Questions: Table 6 represents the list of questions used during the preliminary examination.

A.2. Specific questions: Table 7 represents the list of questions used during evaluations.

Appendix B: Data items extracted from all articles

Table 8 shows the data items extracted from all articles.

Table 8
Data items extracted from all articles

Data Item	Description
Paper identifier	Digital Object Identifier (DOI)
Online Publication Date	Publication Year
Bibliographic Information	Author(s) Name(s), Publication Date, Article Title, and Journal Name
Type of Article	Conference, Workshop and Journal
Motivation	What exactly are the primary aims of this work?
Innovation	Mechanism and context/application
What is the Problem Statement	The problem, as well as a description of it, is addressed and resolved in the research.
What is the method for managing resources?	AI/ML-based Resource Management Technique for Fog/Edge Computing
Implementation Environment	The technique is carried out utilising either a simulated or actual setting.
Performance Evaluation	Which constraints were taken into account when the technique was analysed?
Workload Type	How do you create a dataset for use in experiments?
Performance Metrics	How are the results of a research evaluated using what kind of QoS metrics?
Drawbacks	Where do you see the field of research going in the future?

Appendix C: Journals and Conferences for publishing articles about AI/ML in Fog/Edge Computing.

Table 9 lists the top journals and conferences for publishing articles about AI/ML in fog/edge computing. **Notations:** J – Journal (including IEEE/ACM Transactions), C – Conference, W – Workshop, N – The total number of papers that reported AI/ML-based Resource Management Technique for Fog/Edge Computing as their primary research focus, # – The total number of publications examined.

Appendix D. List of Acronyms

Table 10 shows the list of acronyms.

References

[1] Z. Zhong, M. Xu, M. A. Rodriguez, C. Xu, and R. Buyya, "Machine learning-based orchestration of containers: A taxonomy and future directions," *ACM Computing Surveys (CSUR)*, 2022.

[2] X. Dai, Z. Xiao, H. Jiang, M. Alazab, J. C. Lui, G. Min, S. Dustdar, and J. Liu, "Task offloading for cloud-assisted fog computing with dynamic service caching in enterprise management systems," *IEEE Transactions on Industrial Informatics*, 2022.

[3] S. S. Gill, S. Tuli, M. Xu, I. Singh, K. V. Singh, D. Lindsay, S. Tuli, D. Smirnova, M. Singh, U. Jain *et al.*, "Transformative effects of iot, blockchain and artificial intelligence on cloud computing: Evolution, vision, trends and open challenges," *Internet of Things*, vol. 8, p. 100118, 2019.

[4] A. Hazra, P. K. Donta, T. Amgoth, and S. Dustdar, "Cooperative transmission scheduling and computation offloading with collaboration of fog and cloud for industrial iot applications," *IEEE Internet of Things Journal*, 2022.

[5] A. Chakraborty, M. Kumar *et al.*, "Journey from cloud of things to fog of things: Survey, new trends, and research directions," *Software: Practice and Experience*, oct 2022.

[6] J. Singh *et al.*, "Fog computing: A taxonomy, systematic review, current trends and research challenges," *Journal of Parallel and Distributed Computing*, vol. 157, pp. 56–85, 2021.

[7] M. Sri Raghavendra *et al.*, "Deedsp: Deadline-aware and energy-efficient dynamic service placement in integrated internet of things and fog computing environments," *Transactions on Emerging Telecommunications Technologies*, vol. 32, no. 12, p. e4368, 2021.

[8] V. C. Pujol and S. Dustdar, "Fog robotics—understanding the research challenges," *IEEE Internet Computing*, vol. 25, no. 5, pp. 10–17, 2021.

[9] S. Iftikhar, M. Golec *et al.*, "Fogdlearner: A deep learning-based cardiac health diagnosis framework using fog computing," in *Australasian Computer Science Week 2022*, 2022, pp. 136–144.

[10] V. Karagiannis, P. A. Frangoudis, S. Dustdar, and S. Schulte, "Context-aware routing in fog computing systems," *IEEE Transactions on Cloud Computing*, 2021.

[11] I. Murturi, A. Egyed, and S. Dustdar, "Utilizing ai planning on the edge," *IEEE Internet Computing*, vol. 26, no. 2, pp. 28–35, 2022.

Table 9
Appendix C: Journals, Workshops and Conferences

Publication Venue	J/C/W	#	N
IEEE Transactions on Parallel and Distributed Systems	J	4	4
IEEE Transactions on Cloud Computing	J	6	11
IEEE Transactions on Services Computing	J	4	6
IEEE Internet of Things Journal	J	17	35
IEEE Transactions on Industrial Informatics	J	7	13
IEEE Transactions on Vehicular Technology	J	2	6
IEEE Transactions on Network and Service Management	J	2	3
IEEE Transactions on Sustainable Computing	J	1	3
IEEE/ACM Transactions on Networking	J	1	2
IEEE Transactions on Mobile Computing	J	3	6
IEEE Transactions on Wireless Communications	J	2	5
IEEE Transactions on Green Communications and Networking	J	2	4
IEEE Transactions on Computational Social Systems	J	1	1
IEEE Transactions on Network Science and Engineering	J	2	2
IEEE Transactions on Consumer Electronics	J	1	1
IEEE Transactions on Industry Applications	J	1	1
IEEE Transactions on Broadcasting	J	1	1
IEEE Transactions on Intelligent Transportation Systems	J	2	2
ACM Transactions on Internet Technology	J	4	6
ACM Transactions on Internet of Things	J	4	6
ACM Transactions on Sensor Networks	J	1	1
IEEE Access	J	8	41
Future Generation Computer Systems	J	4	9
Journal of Parallel and Distributed Computing	J	2	4
Journal of Systems and Software	J	4	6
Software: Practice and Experience	J	9	25
Journal of Network and Computer Applications	J	3	5
Transactions on Emerging Telecommunications Technologies	J	1	2
Internet of Things (Elsevier)	J	7	12
IEEE International Symposium on Cluster, Cloud and Internet Computing (CCGrid)	C	1	2
Euromicro Conference on Software Engineering and Advanced Applications	C	1	2
IEEE International Conference on Distributed Computing Systems (ICDCS)	C	2	4
IEEE International Conference on Communications	C	3	5
Australasian Computer Science Week Multiconference	C	2	4
IEEE/ACM International Conference on Utility and Cloud Computing (UCC)	C	1	1
International Conference on Service-Oriented Computing	C	1	3
IEEE International Conference on Pervasive Computing and Communication (PerCom) Workshop	W	1	2
IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)	C	1	2
International Conference on Internet of Things	C	1	4
IEEE International Conference on Networking, Architecture and Storage (NAS)	C	1	2

[12] M. Ghobaei-Arani, A. Souri, and A. A. Rahmanian, "Resource management approaches in fog computing: a comprehensive review," *Journal of Grid Computing*, vol. 18, no. 1, pp. 1–42, 2020.

[13] C. Dehury, S. N. Srirama, P. K. Donta, and S. Dustdar, "Securing clustered edge intelligence with blockchain," *IEEE Consumer Electronics Magazine*, 2022.

[14] A. Yousefpour, C. Fung, T. Nguyen, K. Kadiyala, F. Jalali, A. Niakanlahiji, J. Kong, and J. P. Jue, "All one needs to know about fog computing and related edge computing paradigms: A complete survey," *Journal of Systems Architecture*, vol. 98, pp. 289–330, 2019.

[15] P. Kansal, D. Sharma, and M. Kumar, "Introduction to fog data analytics for iot applications," in *Fog Data Analytics for IoT Applications*. Springer, 2020, pp. 19–38.

[16] A. Y. Ding, E. Peltonen, T. Meuser, A. Aral, C. Becker, S. Dustdar, T. Hiessl, D. Kranzlmüller, M. Liyanage, S. Maghsudi *et al.*, "Roadmap for edge ai: a dagstuhl perspective," pp. 28–33, 2022.

[17] Y. Deng, Z. Chen, D. Zhang, and M. Zhao, "Workload scheduling toward worst-case delay and optimal utility for single-hop fog-iot architecture," *IET Communications*, vol. 12, no. 17, pp. 2164–2173, 2018.

[18] D. Lan, A. Taherkordi, F. Eliassen, L. Liu, S. Delbruel, S. Dustdar, and Y. Yang, "Task partitioning and orchestration on heterogeneous edge platforms: The case of vision applications," *IEEE Internet of Things Journal*, vol. 9, no. 10, pp. 7418–7432, 2022.

Table 10
List of Acronyms

Abbreviation	Description
IoT	Internet of Things
QoS	Quality of Service
SLA	Service-Level Agreement
VM	Virtual Machines
ML	Machine Learning
AI	Artificial Intelligence
SLO	Service Level Objectives
RQ	Research Questions
FC	Fog Computing
EC	Edge Computing
IoHT	Internet of Health Things
PDA	Personal Digital Assistant
IMCF+	IoT Meta-Control Firewall
ITS	Intelligent Transportation Systems
DL	Deep Learning
RL	Reinforcement Learning
DRL	Deep Reinforcement Learning
CL	Centralized learning
FDMA	Frequency Division Multiple Access
TDMA	Time Division Multiple Access
CDC	Cloud Data Centers
VFC	Vehicular Fog Computing
DQN	Deep Q Network
MNO	Mobile Network Operator
FRL	Fuzzy Reinforcement Learning
DDQL	Double Deep Q Learning
MDP	Markov Decision Process
ANN	Artificial Neural Network
GNN	Graph neural network
GPU	Graphical Processing Unit
TPU	Tensor Processing Unit

[19] J. Yang *et al.*, "A federated learning attack method based on edge collaboration via cloud," *Software: Practice and Experience*, pp. 1–18, 2022. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/spe.3180

[20] S. S. Gill, "A manifesto for modern fog and edge computing: Vision, new paradigms, opportunities, and future directions," in *Operationalizing Multi-Cloud Environments*. Springer, 2022, pp. 237–253.

[21] S. S. Gill, M. Xu, C. Ottaviani, P. Patros, R. Bahsoon, A. Shaghaghi, M. Golec, V. Stankovski, H. Wu, A. Abraham *et al.*, "Ai for next generation computing: Emerging trends and future directions," *Internet of Things*, vol. 19, p. 100514, 2022.

[22] S. Tuli *et al.*, "Start: Straggler prediction and mitigation for cloud computing environments using encoder lstm networks," *IEEE Transactions on Services Computing*, 2021.

[23] Y. K. Teoh *et al.*, "Iot and fog computing based predictive maintenance model for effective asset management in industry 4.0 using machine learning," *IEEE Internet of Things Journal*, 2021.

[24] M. Xu *et al.*, "Coscal: Multi-faceted scaling of microservices with reinforcement learning," *IEEE Transactions on Network and Service Management*, 2022.

[25] R. Bianchini, M. Fontoura, E. Cortez, A. Bonde, A. Muzio, A.-M. Constantin, T. Moscibroda, G. Magalhaes, G. Bablani, and M. Russinovich, "Toward ml-centric cloud platforms," *Communications of the ACM*, vol. 63, no. 2, pp. 50–59, 2020.

[26] T. Shao *et al.*, "Iot-pi: A machine learning-based lightweight framework for cost-effective distributed computing using iot," *Internet Technology Letters*, vol. 5, no. 3, p. e355, 2022.

[27] Z. Tang, X. Zhou, F. Zhang, W. Jia, and W. Zhao, "Migration modeling and learning algorithms for containers in fog computing," *IEEE Transactions on Services Computing*, vol. 12, no. 5, pp. 712–725, 2018.

[28] K.-L. A. Yau, J. Qadir, H. L. Khoo, M. H. Ling, and P. Komisarczuk, "A survey on reinforcement learning models and algorithms for traffic signal control," *ACM Computing Surveys (CSUR)*, vol. 50, no. 3, pp. 1–38, 2017.

[29] T. L. Duc, R. G. Leiva, P. Casari, and P.-O. Östberg, "Machine learning methods for reliable resource provisioning in edge-cloud computing: A survey," *ACM Computing Surveys (CSUR)*, vol. 52, no. 5, pp. 1–39, 2019.

[30] E. Casalicchio, "Container orchestration: a survey," *Systems Modeling: Methodologies and Tools*, pp. 221–235, 2019.

[31] S. Deng, H. Zhao, W. Fang, J. Yin, S. Dustdar, and A. Y. Zomaya, "Edge intelligence: The confluence of edge computing and artificial intelligence," *IEEE Internet of Things Journal*, vol. 7, no. 8, pp. 7457–7469, 2020.

[32] P. Kansal, M. Kumar, and O. P. Verma, "Classification of resource management approaches in fog/edge paradigm and future research prospects: a systematic review," *The Journal of Supercomputing*, vol. 78, no. 11, pp. 13 145–13 204, 2022.

[33] B. Kitchenham, "Procedures for performing systematic reviews," *Keele, UK, Keele University*, vol. 33, no. 2004, pp. 1–26, 2004.

[34] X. Yang and N. Rahmani, "Task scheduling mechanisms in fog computing: review, trends, and perspectives," *Kybernetes*, 2020.

[35] K. H. Abdulkareem, M. A. Mohammed, S. S. Gunasekaran, M. N. Al-Mhiqani, A. A. Mutlag, S. A. Mostafa, N. S. Ali, and D. A. Ibrahim, "A review of fog computing and machine learning: concepts, applications, challenges, and open issues," *IEEE Access*, vol. 7, pp. 153 123–153 140, 2019.

[36] H. F. Atlam, R. J. Walters, and G. B. Wills, "Fog computing and the internet of things: A review," *big data and cognitive computing*, vol. 2, no. 2, p. 10, 2018.

[37] Z. M. Nayeri, T. Ghafarian, and B. Javadi, "Application placement in fog computing with ai approach: Taxonomy and a state of the art survey," *Journal of Network and Computer Applications*, vol. 185, p. 103078, 2021.

[38] H. Tran-Dang, S. Bhardwaj, T. Rahim, A. Musaddiq, and D.-S. Kim, "Reinforcement learning based resource management for fog computing environment: Literature review, challenges, and open issues," *Journal of Communications and Networks*, 2022.

[39] S. Askar, Z. J. Hamad, and S. W. Kareem, "Deep learning and fog computing: A review," *International Journal of Science and Business*, vol. 5, no. 6, pp. 197–208, 2021.

[40] N. Kumari, A. Yadav, and P. K. Jana, "Task offloading in fog computing: A survey of algorithms and optimization techniques," *Computer Networks*, vol. 214, p. 109137, 2022.

[41] S. S. Gill, R. C. Arya, G. S. Wander, and R. Buyya, "Fog-based smart healthcare as a big data and cloud service for heart patients using iot," in *International Conference on Intelligent Data Communication Technologies and Internet of Things*. Springer, 2018, pp. 1376–1383.

[42] A. N Toosi, C. Agarwal, L. Mashayekhy, S. K. Moghaddam, R. Mahmud, and Z. Tari, "Greenfog: A framework for sustainable fog computing," in *International Conference on Service-Oriented Computing*. Springer, 2022, pp. 540–549.

[43] B. Jennings and R. Stadler, "Resource management in clouds: Survey and research challenges," *Journal of Network and Systems Management*, vol. 23, no. 3, pp. 567–619, 2015.

[44] S. Tuli *et al.*, "Healthfog: An ensemble deep learning based smart healthcare system for automatic diagnosis of heart diseases in integrated iot and fog computing environments," *Future Generation Computer Systems*, vol. 104, pp. 187–200, 2020.

[45] İ. Kök, F. Y. Okay, and S. Özdemir, "Fogai: An ai-supported fog controller for next generation iot," *Internet of Things*, vol. 19, p. 100572, 2022.

[46] A. Shakarami, H. Shakarami, M. Ghobaei-Arani, E. Nikougoftar, and M. Faraji-Mehmandar, "Resource provisioning in edge/fog computing: A comprehensive and systematic review," *Journal of Systems Architecture*, vol. 122, p. 102362, 2022.

[47] S. Iftikhar *et al.*, "Tesco: Multiple simulations based ai-augmented fog computing for qos optimization," in *The 22nd IEEE International Conference on Scalable Computing and Communications (ScalCom 2022), Hainan, China, 15-18 December 2022*, 2022.

[48] D. Lindsay *et al.*, "The evolution of distributed computing systems: from fundamental to new frontiers," *Computing*, vol. 103, no. 8, pp. 1859–1878, 2021.

[49] S. S. Gill *et al.*, "Router: Fog enabled cloud based intelligent resource management approach for smart home iot devices," *Journal of Systems and Software*, vol. 154, pp. 125–138, 2019.

[50] S. Tuli *et al.*, "Hunter: Ai based holistic resource management for sustainable cloud computing," *Journal of Systems and Software*, vol. 184, p. 111124, 2022.

[51] S. S. Nabavi *et al.*, "Tractor: Traffic-aware and power-efficient virtual machine placement in edge-cloud data centers using artificial bee colony optimization," *International Journal of Communication Systems*, vol. 35, no. 1, p. e4747, 2022.

[52] A. Souri, "Artificial intelligence mechanisms for management of qos-aware connectivity in internet of vehicles," *Journal of High Speed Networks*, no. Preprint, pp. 1–10, 2022.

[53] S. Iftikhar *et al.*, "Hunterplus: Ai based energy-efficient task scheduling for cloud-fog computing environments," *Internet of Things*, p. 100667, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2542660522001482

[54] A. Souri, M.-Y. Chen, and N. J. Navimipour, "Computational intelligence methods for smart connectivity in iot," p. 2202001, 2022.

[55] S. Tuli *et al.*, "Next generation technologies for smart healthcare: Challenges, vision, model, trends and future directions," *Internet technology letters*, vol. 3, no. 2, p. e145, 2020.

[56] H. Habibzadeh, K. Dinesh, O. R. Shishvan, A. Boggio-Dandry, G. Sharma, and T. Soyata, "A survey of healthcare internet of things (hiot): A clinical perspective," *IEEE Internet of Things Journal*, vol. 7, no. 1, pp. 53–71, 2019.

[57] T. Wu, F. Wu, C. Qiu, J.-M. Redouté, and M. R. Yuce, "A rigid-flex wearable health monitoring sensor patch for iot-connected healthcare applications," *IEEE Internet of Things Journal*, vol. 7, no. 8, pp. 6932–6945, 2020.

[58] S. Esmaeili, S. R. K. Tabbakh, and H. Shakeri, "A priority-aware lightweight secure sensing model for body area networks with clinical healthcare applications in internet of things," *Pervasive and Mobile Computing*, vol. 69, p. 101265, 2020.

[59] W. Huifeng, S. N. Kadry, and E. D. Raj, "Continuous health monitoring of sportsperson using iot devices based wearable technology," *Computer Communications*, vol. 160, pp. 588–595, 2020.

[60] C. M. Dourado, S. P. P. da Silva, R. V. M. da Nobrega, P. P. Reboucas Filho, K. Muhammad, and V. H. C. de Albuquerque, "An open ioht-based deep learning framework for online medical image recognition," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 2, pp. 541–548, 2020.

[61] P. P. Ray, N. Thapa, D. Dash, and D. De, "Novel implementation of iot based non-invasive sensor system for real-time monitoring of intravenous fluid level for assistive e-healthcare," *Circuit World*, vol. 45, no. 3, pp. 109–123, 2019.

[62] J. Das, S. Ghosh, A. Mukherjee, S. K. Ghosh, and R. Buyya, "Rescue: Enabling green healthcare services using integrated iot-edge-fog-cloud computing environments," *Software: Practice and Experience*, 2022.

[63] A. Kumar *et al.*, "A drone-based networked system and methods for combating coronavirus disease (covid-19) pandemic," *Future Generation Computer Systems*, vol. 115, pp. 1–19, 2021.

[64] S. Tuli *et al.*, "Predicting the growth and trend of covid-19 pandemic using machine learning and cloud computing," *Internet of Things*, vol. 11, p. 100222, 2020.

[65] N. Khan, Z. Ma, A. Ullah, and K. Polat, "Dca-iomt: Knowledge graph embedding-enhanced deep collaborative alerts-recommendation against covid19," *IEEE Transactions on Industrial Informatics*, 2022.

[66] X. Lin, J. Wu, A. K. Bashir, W. Yang, A. Singh, and A. A. AlZubi, "Fairhealth: Long-term proportional fairness-driven 5g edge healthcare in internet of medical things," *IEEE Transactions on Industrial Informatics*, 2022.

[67] F. Desai *et al.*, "Healthcloud: A system for monitoring health status of heart patients using machine learning and cloud computing," *Internet of Things*, vol. 17, p. 100485, 2022.

[68] N. K. Dewangan and P. Chandrakar, "Patient-centric token-based healthcare blockchain implementation using secure internet of medical things," *IEEE Transactions on Computational Social Systems*, 2022.

[69] W. Lv, S. Wu, C. Jiang, Y. Cui, X. Qiu, and Y. Zhang, "Towards large-scale and privacy-preserving contact tracing in covid-19 pandemic: a blockchain perspective," *IEEE Transactions on Network Science and Engineering*, 2020.

[70] S. Kumar, R. D. Raut, P. Priyadarshinee, S. K. Mangla, U. Awan, and B. E. Narkhede, "The impact of iot on the performance of vaccine supply chain distribution in the covid-19 context," *IEEE Transactions on Engineering Management*, 2022.

[71] C. Gavrila, V. Popescu, M. Fadda, M. Anedda, and M. Murroni, "On the suitability of hbbtv for unified smart home experience," *IEEE Transactions on Broadcasting*, vol. 67, no. 1, pp. 253–262, 2020.

[72] A. Chatterjee, S. Paul, and B. Ganguly, "Multi-objective energy management of a smart home in real time environment," *IEEE Transactions on Industry Applications*, 2022.

[73] M. Yamauchi, Y. Ohsita, M. Murata, K. Ueda, and Y. Kato, "Anomaly detection in smart home operation from user behaviors and home conditions," *IEEE Transactions on Consumer Electronics*, vol. 66, no. 2, pp. 183–192, 2020.

[74] A. K. Sikder, L. Babun, Z. B. Celik, H. Aksu, P. McDaniel, E. Kirda, and A. S. Uluagac, "Who's controlling my device? multi-user multi-device-aware access control system for shared smart home environment," *ACM Transactions on Internet of Things*, vol. 3, no. 4, pp. 1–39, 2022.

[75] X. Li and D. Li, "Gpfs: a graph-based human pose forecasting system for smart home with online learning," *ACM Transactions on Sensor Networks (TOSN)*, vol. 17, no. 3, pp. 1–19, 2021.

[76] S. Constantinou, A. Konstantinidis, P. K. Chrysanthis, and D. Zeinalipour-Yazti, "Green planning of iot home automation workflows in smart buildings," *ACM Transactions on Internet of Things*, vol. 3, no. 4, pp. 1–30, 2022.

[77] Y. Liu, X. Ma, L. Shu, G. P. Hancke, and A. M. Abu-Mahfouz, "From industry 4.0 to agriculture 4.0: Current status, enabling technologies, and research challenges," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 6, pp. 4322–4334, 2020.

[78] M. E. E. Alahi, L. Xie, S. Mukhopadhyay, and L. Burkitt, "A temperature compensated smart nitrate-sensor for agricultural industry," *IEEE Transactions on industrial electronics*, vol. 64, no. 9, pp. 7333–7341, 2017.

[79] A. Sengupta *et al.*, "Mobile edge computing based internet of agricultural things: A systematic review and future directions," *Mobile Edge Computing*, pp. 415–441, 2021.

[80] T. Wang, X. Wang, Y. Jiang, Z. Sun, Y. Liang, X. Hu, H. Li, Y. Shi, J. Xu, and J. Ruan, "Hybrid machine learning approach for evapotranspiration estimation of fruit tree in agricultural cyber-physical systems," *IEEE Transactions on Cybernetics*, 2022.

[81] S. Singh, I. Chana, and R. Buyya, "Agri-info: cloud based autonomic system for delivering agriculture as a service," *Internet of Things*, vol. 9, p. 100131, 2020.

[82] J. Bauer and N. Aschenbruck, "Towards a low-cost rssi-based crop monitoring," *ACM Transactions on Internet of Things*, vol. 1, no. 4, pp. 1–26, 2020.

[83] S. S. Gill, I. Chana, and R. Buyya, "Iot based agriculture as a cloud and big data service: the beginning of digital india," *Journal of Organizational and End User Computing (JOEUC)*, vol. 29, no. 4, pp. 1–23, 2017.

[84] W.-J. Hu, J. Fan, Y.-X. Du, B.-S. Li, N. Xiong, and E. Bekkering, "Mdfc–resnet: an agricultural iot system to accurately recognize crop diseases," *IEEE Access*, vol. 8, pp. 115 287–115 298, 2020.

[85] F. Zhu, Y. Lv, Y. Chen, X. Wang, G. Xiong, and F.-Y. Wang, "Parallel transportation systems: Toward iot-enabled smart urban traffic control and management," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 10, pp. 4063–4071, 2019.

[86] R. Ke, Y. Zhuang, Z. Pu, and Y. Wang, "A smart, efficient, and reliable parking surveillance system with edge artificial intelligence on iot devices," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 8, pp. 4962–4974, 2020.

[87] K. Bansal *et al.*, "Deepbus: Machine learning based real time pothole detection system for smart transportation using iot," *Internet Technology Letters*, vol. 3, no. 3, p. e156, 2020.

[88] B. V. Philip, T. Alpcan, J. Jin, and M. Palaniswami, "Distributed real-time iot for autonomous vehicles," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 2, pp. 1131–1140, 2018.

[89] S. Chavhan, D. Gupta, S. P. Gochhayat, C. B. N, A. Khanna, K. Shankar, and J. J. P. C. Rodrigues, "Edge computing ai-iot integrated energy efficient intelligent transportation system for smart cities," *ACM Trans. Internet Technol.*, dec 2021, just Accepted. [Online]. Available: https://doi.org/10.1145/3507906

[90] L. Wan, M. Zhang, L. Sun, and X. Wang, "Machine learning empowered iot for intelligent vehicle location in smart cities," *ACM Transactions on Internet Technology (TOIT)*, vol. 21, no. 3, pp. 1–25, 2021.

[91] F. Piccialli, F. Giampaolo, E. Prezioso, D. Crisci, and S. Cuomo, "Predictive analytics for smart parking: A deep learning approach in forecasting of iot data," *ACM Transactions on Internet Technology (TOIT)*, vol. 21, no. 3, pp. 1–21, 2021.

[92] K. A. Eldrandaly, M. Abdel-Basset, and L. A. Shawky, "Internet of spatial things: A new reference model with insight analysis," *IEEE Access*, vol. 7, pp. 19 653–19 669, 2019.

[93] M. Sarwat, "Spatial data systems support for the internet of things: challenges and opportunities," *Sigspatial Special*, vol. 12, no. 2, pp. 42–47, 2020.

[94] S. Ghosh, A. Mukherjee, S. K. Ghosh, and R. Buyya, "Mobi-iost: mobility-aware cloud-fog-edge-iot collaborative framework for time-critical applications," *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 4, pp. 2271–2285, 2019.

[95] J. Y. Koh, I. Nevat, D. Leong, and W.-C. Wong, "Geo-spatial location spoofing detection for internet of things," *IEEE Internet of Things Journal*, vol. 3, no. 6, pp. 971–978, 2016.

[96] S. Ghosh, A. Mukherjee, S. Ghosh, and R. Buyya, "Stoppage: Spatio-temporal data driven cloud-fog-edge computing framework for pandemic monitoring and management," *Software: Practice and Experience*, 2022.

[97] M. Golec *et al.*, "Biosec: A biometric authentication framework for secure and private communication among edge devices in iot and industry 4.0," *IEEE Consumer Electronics Magazine*, vol. 11, no. 2, pp. 51–56, 2022.

[98] L. S. Vailshery, "Iot connected devices worldwide 2019-2030," Aug 2022. [Online]. Available: https://www.statista.com/statistics/1183457/iot-connected-devices-worldwide/

[99] M. Golec *et al.*, "Aiblock: Blockchain based lightweight framework for serverless computing using ai," in *2022 22nd IEEE International Symposium on Cluster, Cloud and Internet Computing (CCGrid)*. IEEE, 2022, pp. 886–892.

[100] M. Golec, R. Ozturac *et al.*, "ifaasbus: A security-and privacy-based lightweight framework for serverless computing using iot and machine learning," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 5, pp. 3522–3529, 2021.

[101] M. S. Miah, M. Schukat, and E. Barrett, "An enhanced sum rate in the cluster based cognitive radio relay network using the sequential approach for the future internet of things," *Human-centric Computing and Information Sciences*, vol. 8, no. 1, pp. 1–27, 2018.

[102] S. Iftikhar, M. Golec *et al.*, "Fog computing based router-distributor application for sustainable smart home," in *2022 IEEE 95th Vehicular Technology Conference:(VTC2022-Spring)*. IEEE, 2022, pp. 1–5.

[103] J. Guo, C. Li, Y. Chen, and Y. Luo, "On-demand resource provision based on load estimation and service expenditure in edge cloud environment," *Journal of network and computer applications*, vol. 151, p. 102506, 2020.

[104] L. Li, K. Ota, , and M. Dong, "Humanlike driving: Empirical decisionmaking system for autonomous vehicles," *IEEE Trans. Veh. Technol.*, vol. 67, no. 8, pp. 6814–6823, 2018.

[105] Q. Wang, Y. Guo, L. Yu, and P. Li, "Earthquake prediction based on spatio-temporal data mining: An lstm network approach," *IEEE Trans. Emerg. Topics Comput.*, vol. 8, no. 1, p. 148–158, 2020.

[106] S. U. Amin, M. S. Hossain, G. Muhammad, M. Alhussein, and M. A. Rahman, "Cognitive smart healthcare for pathology detection and monitoring," *IEEE Access*, vol. 7, pp. 10 745–10 753, 2019.

[107] H. Xu, C.-Y. Ho, A. M. Abdelmoniem, A. Dutta, E. H. Bergou, K. Karatsenidis, M. Canini, and P. Kalnis, "Grace: A compressed communication framework for distributed machine learning," in *2021 IEEE 41st International Conference on Distributed Computing Systems (ICDCS)*, 2021, pp. 561–572.

[108] A. M. Abdelmoniem and M. Canini, "Dc2: Delay-aware compression control for distributed machine learning," in *IEEE Conference on Computer Communications (INFOCOM)*. IEEE, 2021, pp. 1–10.

[109] A. M Abdelmoniem, A. Elzanaty, M.-S. Alouini, and M. Canini, "An efficient statistical-based gradient compression technique for distributed training systems," *Proceedings of Machine Learning and Systems (MLSys)*, vol. 3, pp. 297–322, 2021.

[110] S. Misra, A. K. Tyagi, V. Piuri, and L. Garg, *Artificial Intelligence for Cloud and Edge Computing*. Springer Nature, 2022.

[111] L. Wang, L. Jiao, J. Li, and M. Mühlhäuser, "Online resource allocation for arbitrary user mobility in distributed edge clouds," in *IEEE International Conference on Distributed Computing Systems (ICDCS)*, 2017.

[112] D. Rosendo, A. Costan, P. Valduriez, and G. Antoniu, "Distributed intelligence on the edge-to-cloud continuum: A systematic literature review," *Journal of Parallel and Distributed Computing*, vol. 166, pp. 71–94, 2022.

[113] D. T. Nguyen, L. B. Le, and V. K. Bhargava, "A market-based framework for multi-resource allocation in fog computing," *IEEE/ACM Transactions on Networking*, vol. 27, no. 3, pp. 1151–1164, 2019.

[114] K. Bonawitz, H. Eichner, W. Grieskamp, D. Huba, A. Ingerman, V. Ivanov, C. Kiddon, J. Konečnỳ, S. Mazzocchi, B. McMahan *et al.*, "Towards federated learning at scale: System design," *Proceedings of Machine Learning and Systems*, vol. 1, pp. 374–388, 2019.

[115] H. Daga, P. K. Nicholson, A. Gavrilovska, and D. Lugones, "Cartel: A system for collaborative transfer learning at the edge," in *Proceedings of the ACM Symposium on Cloud Computing*, 2019, pp. 25–37.

[116] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings *et al.*, "Advances and open problems in federated learning," *Foundations and Trends® in Machine Learning*, vol. 14, no. 1–2, pp. 1–210, 2021.

[117] A. M. Abdelmoniem, C.-Y. Ho, P. Papageorgiou, and M. Canini, "Empirical analysis of federated learning in heterogeneous environments," in *ACM EuroMLSys*, 2022.

[118] A. M. Abdelmoniem, A. N. Sahu, M. Canini, and S. A. Fahmy, "Resource-efficient federated learning," *ArXiv abs/2111.01108*, 2021.

[119] S. Zarandi and H. Tabassum, "Federated double deep q-learning for joint delay and energy minimization in iot networks," *ArXiv 2104.11320*, 2021.

[120] R. Fantacci and B. Picano, "Federated learning framework for mobile edge computing networks," *CAAI Transactions on Intelligence Technology*, vol. 5, no. 1, pp. 15–21, 2020.

[121] X. Chen and G. Liu, "Federated deep reinforcement learning-based task offloading and resource allocation for smart cities in a mobile edge network," *Sensors*, vol. 22, no. 13, 2022.

[122] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial intelligence and statistics*. PMLR, 2017, pp. 1273–1282.

[123] A. M. Abdelmoniem and M. Canini, "Towards mitigating device heterogeneity in federated learning via adaptive model quantization," in *ACM EuroMLSys*, 2021.

[124] J. McChesney, N. Wang, A. Tanwer, E. De Lara, and B. Varghese, "Defog: fog computing benchmarks," in *Proceedings of the 4th ACM/IEEE Symposium on Edge Computing*, 2019, pp. 47–58.

[125] Ó. Fontenla-Romero, B. Guijarro-Berdiñas, D. Martinez-Rego, B. Pérez-Sánchez, and D. Peteiro-Barral, "Online machine learning," in *Efficiency and Scalability Methods for Computational Intellect*. IGI Global, 2013, pp. 27–54.

[126] E. Bisong, "Batch vs. online learning," in *Building Machine Learning and Deep Learning Models on Google Cloud Platform*. Springer, 2019, pp. 199–201.

[127] S. A. Moqurrab, N. Tariq *et al.*, "A deep learning-based privacy-preserving model for smart healthcare in internet of medical things using fog computing," *Wireless Personal Communications*, vol. 126, no. 3, pp. 2379–2401, 2022.

[128] P. Singh *et al.*, "Machine learning for cloud, fog, edge and serverless computing environments: comparisons, performance evaluation benchmark and future directions," *International Journal of Grid and Utility Computing*, vol. 13, no. 4, pp. 447–457, 2022.

[129] A. Dhillon, A. Singh *et al.*, "Iotpulse: machine learning-based enterprise health information system to predict alcohol addiction in punjab (india) using iot and fog computing," *Enterprise Information Systems*, vol. 16, no. 7, p. 1820583, 2022.

[130] S. Tuli and Others, "ithermofog: Iot-fog based automatic thermal profile creation for cloud data centers using artificial intelligence techniques," *Internet Technology Letters*, vol. 3, no. 5, p. e198, 2020.

[131] A. Yousefpour, C. Fung, T. Nguyen, K. Kadiyala, F. Jalali, A. Niakanlahiji, J. Kong, and J. P. Jue, "All one needs to know about fog computing and related edge computing paradigms: A complete survey," *Journal of Systems Architecture*, vol. 98, pp. 289–330, 2019.

[132] S. Ghafouri, A. Karami *et al.*, "Mobile-kube: Mobility-aware and energy-efficient service orchestration on kubernetes edge servers," in *15th IEEE/ACM International Conference on Utility and Cloud Computing (UCC 2022), Washington State University Portland, OR, United States, December 6-9, 2022*, 2022.

[133] M. Sriraghavendra *et al.*, "Dosp: A deadline-aware dynamic service placement algorithm for workflow-oriented iot applications in fog-cloud computing environments," in *Energy Conservation Solutions for Fog-Edge Computing Paradigms*. Springer, 2022, pp. 21–47.

[134] S. S. Gill, "Quantum and blockchain based serverless edge computing: A vision, model, new trends and future directions," *Internet Technology Letters*, p. e275, 2021.

[135] T. Klervie and N. T. Simin, "A taxonomy for management and optimization of multiple resources in edge computing," *Wireless Communications & Mobile Computing*, 2018.

[136] M. Aazam and E.-N. Huh, "Dynamic resource provisioning through fog micro datacenter," in *2015 IEEE International Conference on Pervasive Computing and Communication Workshops (PerCom Workshops)*, 2015, pp. 105–110.

[137] M. Aazam and E.-N. a. Huh, "Fog computing micro datacenter based dynamic resource estimation and pricing model for iot," in *2015 IEEE 29th International Conference on Advanced Information Networking and Applications*. IEEE, 2015, pp. 687–694.

[138] H. G. Myung, J. Lim, and D. J. Goodman, "Single carrier fdma for uplink wireless transmission," *IEEE vehicular technology magazine*, vol. 1, no. 3, pp. 30–38, 2006.

[139] I. Rhee, A. Warrier, J. Min, and L. Xu, "Drand: Distributed randomized tdma scheduling for wireless ad hoc networks," *IEEE Transactions on Mobile Computing*, vol. 8, no. 10, pp. 1384–1396, 2009.

[140] P. Athwani and D. P. Vidyarthi, "Resource discovery in mobile cloud computing: A clustering based approach," in *2015 IEEE UP Section Conference on Electrical Computer and Electronics (UPCON)*. IEEE, 2015, pp. 1–6.

[141] H. R. Arkian, R. E. Atani, A. Diyanat, and A. Pourkhalili, "A cluster-based vehicular cloud architecture with learning-based resource management," *The Journal of Supercomputing*, vol. 71, no. 4, pp. 1401–1426, 2015.

[142] W. Liu, T. Nishio, R. Shinkuma, and T. Takahashi, "Adaptive resource discovery in mobile cloud computing," *Computer Communications*, vol. 50, pp. 119–129, 2014.

[143] K. M. Sim, "Agent-based fog computing: Gossiping, reasoning, and bargaining," *IEEE Letters of the Computer Society*, vol. 1, no. 2, pp. 21–24, 2018.

[144] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural networks*, vol. 61, pp. 85–117, 2015.

[145] R. Yadav and G. Baranwal, "Trust-aware framework for application placement in fog computing," in *2019 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*. IEEE, 2019, pp. 1–6.

[146] G. Li, J. Song, J. Wu, and J. Wang, "Method of resource estimation based on qos in edge computing," *Wireless Communications & Mobile Computing*, vol. 2018, pp. 1–9, 2018.

[147] F. Saeik, M. Avgeris, D. Spatharakis, N. Santi, D. Dechouniotis, J. Violos, A. Leivadeas, N. Athanasopoulos, N. Mitton, and S. Papavassiliou, "Task offloading in edge and cloud computing: A survey on mathematical, artificial intelligence and control theory solutions," *Computer Networks*, vol. 195, p. 108177, 2021.

[148] C. You, K. Huang, and H. Chae, "Energy efficient mobile cloud computing powered by wireless energy transfer," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 5, pp. 1757–1771, 2016.

[149] S. Bi and Y. J. Zhang, "Computation rate maximization for wireless powered mobile-edge computing with binary computation offloading," *IEEE Transactions on Wireless Communications*, vol. 17, no. 6, pp. 4177–4190, 2018.

[150] L. Shi, Y. Ye, X. Chu, and G. Lu, "Computation energy efficiency maximization for a noma-based wpt-mec network," *IEEE Internet of Things Journal*, vol. 8, no. 13, pp. 10 731–10 744, 2021.

[151] C. You, K. Huang, H. Chae, and B. H. Kim, "Energy-efficient resource allocation for mobile-edge computation offloading," *IEEE Transactions on Wireless Communications*, vol. 16, no. 3, pp. 1397–1411, 2017.

[152] P.-Q. Huang, Y. Wang, K. Wang, and Q. Zhang, "Combining lyapunov optimization with evolutionary transfer optimization for long-term energy minimization in irs-aided communications," *IEEE Transactions on Cybernetics*, pp. 1–11, 2022.

[153] L. Shi, Y. Ye, X. Chu, and G. Lu, "Computation bits maximization in a backscatter assisted wirelessly powered mec network," *IEEE Communications Letters*, vol. 25, no. 2, pp. 528–532, 2021.

[154] J. Yao and N. Ansari, "Task allocation in fog-aided mobile iot by lyapunov online reinforcement learning," *IEEE Transactions on Green Communications and Networking*, vol. 4, no. 2, pp. 556–565, 2020.

[155] J. Lu, L. Chen, J. Xia, F. Zhu, M. Tang, C. Fan, and J. Ou, "Analytical offloading design for mobile edge computing-based smart internet of vehicle," *EURASIP journal on advances in signal processing*, vol. 2022, no. 1, pp. 1–19, 2022.

[156] M. Zeng and V. Fodor, "Dynamic spectrum sharing for load balancing in multi-cell mobile edge computing," *IEEE Wireless Communications Letters*, vol. 9, no. 2, pp. 189–193, 2019.

[157] A. Thomas, G. Krishnalal, and V. J. Raj, "Credit based scheduling algorithm in cloud computing environment," *Procedia Computer Science*, vol. 46, pp. 913–920, 2015.

[158] B. Mondal, K. Dasgupta, and P. Dutta, "Load balancing in cloud computing using stochastic hill climbing-a soft computing approach," *Procedia Technology*, vol. 4, no. 1, pp. 783–789, 2012.

[159] S. U. Malik *et al.*, "Effort: Energy efficient framework for offload communication in mobile cloud computing," *Software: Practice and Experience*, vol. 51, no. 9, pp. 1896–1909, 2021.

[160] D.-N. Vu, N.-N. Dao, W. Na, and S. Cho, "Dynamic resource orchestration for service capability maximization in fog-enabled connected vehicle networks," *IEEE Transactions on Cloud Computing*, vol. 10, no. 3, pp. 1726–1737, 2022.

[161] M. Min, L. Xiao, Y. Chen, P. Cheng, D. Wu, and W. Zhuang, "Learning-based computation offloading for iot devices with energy harvesting," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 2, pp. 1930–1941, 2019.

[162] T. L. Duc, R. G. Leiva, P. Casari, and P.-O. Östberg, "Machine learning methods for reliable resource provisioning in edge-cloud computing: A survey," *ACM Computing Surveys (CSUR)*, vol. 52, no. 5, pp. 1–39, 2019.

[163] M. Arif, A. K. Kiani, and J. Qadir, "Machine learning based optimized live virtual machine migration over wan links," *Telecommunication Systems*, vol. 64, no. 2, pp. 1–13, 2016.

[164] P. Cunningham, M. Cord, and S. J. Delany, "Supervised learning," in *Machine learning techniques for multimedia*. Springer, 2008, pp. 21–49.

[165] V. Nasteski, "An overview of the supervised machine learning methods," *Horizons. b*, vol. 4, pp. 51–62, 2017.

[166] B. Cao, L. Zhang, Y. Li, D. Feng, and W. Cao, "Intelligent offloading in multi-access edge computing: A state-of-the-art review and framework," *IEEE Communications Magazine*, vol. 57, no. 3, pp. 56–62, 2019.

[167] H. B. Barlow, "Unsupervised learning," *Neural computation*, vol. 1, no. 3, pp. 295–311, 1989.

[168] J. Han and Z. Ge, "Effect of dimensionality reduction on stock selection with cluster analysis in different market situations," *Expert Systems with Applications*, vol. 147, p. 113226, 2020.

[169] M. Dash, H. Liu, and J. Yao, "Dimensionality reduction of unsupervised data," in *Proceedings ninth ieee international conference on tools with artificial intelligence*. IEEE, 1997, pp. 532–539.

[170] A. Kassambara, *Practical guide to cluster analysis in R: Unsupervised machine learning*. Sthda, 2017, vol. 1.

[171] G. Carvalho, B. Cabral, V. Pereira, and J. Bernardino, "Computation offloading in edge computing environments using artificial intelligence techniques," *Engineering Applications of Artificial Intelligence*, vol. 95, p. 103840, 2020.

[172] T. Chen, X. Zhang, M. You, G. Zheng, and S. Lambotharan, "A gnn-based supervised learning framework for resource allocation in wireless iot networks," *IEEE Internet of Things Journal*, vol. 9, no. 3, pp. 1712–1724, 2021.

[173] Z. Wang, M. Eisen, and A. Ribeiro, "Learning decentralized wireless resource allocations with graph neural networks," *IEEE Transactions on Signal Processing*, vol. 70, pp. 1850–1863, 2022.

[174] T. Yang, G. Andrew, H. Eichner, H. Sun, W. Li, N. Kong, D. Ramage, and F. Beaufays, "Applied Federated Learning: Improving Google Keyboard Query Suggestions," *arXiv 1812.02903*, 2018.

[175] X. Chen, H. Zhang, C. Wu, S. Mao, Y. Ji, and M. Bennis, "Optimized computation offloading performance in virtual edge computing systems via deep reinforcement learning," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4005–4018, 2018.

[176] S. Vemireddy and R. R. Rout, "Fuzzy reinforcement learning for energy efficient task offloading in vehicular fog computing," *Computer Networks*, vol. 199, p. 108463, 2021.

[177] S. Tuli, G. Casale, and N. R. Jennings, "Gosh: Task scheduling using deep surrogate models in fog computing environments," *IEEE Transactions on Parallel and Distributed Systems*, vol. 33, no. 11, pp. 2821–2833, 2021.

[178] S. Tuli, G. Casale, and N. R. a. Jennings, "Mcds: Ai augmented workflow scheduling in mobile edge cloud computing systems," *IEEE Transactions on Parallel and Distributed Systems*, vol. 33, no. 11, pp. 2794–2807, 2021.

[179] M. Chen, Y. Xiao, Q. Li, and K.-c. Chen, "Minimizing age-of-information for fog computing-supported vehicular networks with deep q-learning," in *ICC 2020-2020 IEEE International Conference on Communications (ICC)*. IEEE, 2020, pp. 1–6.

[180] P. Gazori, D. Rahbari, and M. Nickray, "Saving time and cost on the scheduling of fog-based iot applications using deep reinforcement learning approach," *Future Generation Computer Systems*, vol. 110, pp. 1098–1115, 2020.

[181] J. Shi, J. Du, J. Wang, J. Wang, and J. Yuan, "Priority-aware task offloading in vehicular fog computing based on deep reinforcement learning," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 12, pp. 16 067–16 081, 2020.

[182] Z. Ning, P. Dong, X. Wang, J. J. Rodrigues, and F. Xia, "Deep reinforcement learning for vehicular edge computing: An intelligent offloading system," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 10, no. 6, pp. 1–24, 2019.

[183] S. Conti, G. Faraci, R. Nicolosi, S. A. Rizzo, and G. Schembra, "Battery management in a green fog-computing node: A reinforcement-learning approach," *IEEE Access*, vol. 5, pp. 21 126–21 138, 2017.

[184] T. Yang, Y. Hu, M. C. Gursoy, A. Schmeink, and R. Mathar, "Deep reinforcement learning based resource allocation in low latency edge computing networks," in *2018 15th international symposium on wireless communication systems (ISWCS)*. IEEE, 2018, pp. 1–5.

[185] J. Chen, S. Chen, Q. Wang, B. Cao, G. Feng, and J. Hu, "iraf: A deep reinforcement learning approach for collaborative mobile edge computing iot networks," *IEEE Internet of Things Journal*, vol. 6, no. 4, pp. 7011–7024, 2019.

[186] X. Liu, Z. Qin, and Y. Gao, "Resource allocation for edge computing in iot networks via reinforcement learning," in *ICC 2019-2019 IEEE international conference on communications (ICC)*. IEEE, 2019, pp. 1–6.

[187] M. S. Aslanpour *et al.*, "Performance evaluation metrics for cloud, fog and edge computing: A review, taxonomy, benchmarks and standards for future research," *Internet of Things*, vol. 12, p. 100273, 2020.

[188] B. Costa, P. F. Pires, and F. C. Delicato, "Modeling iot applications with sysml4iot," in *2016 42th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*. IEEE, 2016, pp. 157–164.

[189] X. Zeng, S. K. Garg, P. Strazdins, P. P. Jayaraman, D. Georgakopoulos, and R. Ranjan, "Iotsim: A simulator for analysing iot applications," *Journal of Systems Architecture*, vol. 72, pp. 93–107, 2017.

[190] C. Mechalikh, H. Taktak, and F. Moussa, "Pureedgesim: A simulation toolkit for performance evaluation of cloud, fog, and pure edge computing environments," in *2019 International Conference on High Performance Computing & Simulation (HPCS)*. IEEE, 2019, pp. 700–707.

[191] D. N. Jha, K. Alwasel, A. Alshoshan, X. Huang, R. K. Naha, S. K. Battula, S. Garg, D. Puthal, P. James, A. Zomaya *et al.*, "Iotsim-edge: a simulation framework for modeling the behavior of internet of things and edge computing environments," *Software: Practice and Experience*, vol. 50, no. 6, pp. 844–867, 2020.

[192] C. Wang, R. Li, W. Li, C. Qiu, and X. Wang, "Simedgeintel: A open-source simulation platform for resource management in edge intelligence," *Journal of Systems Architecture*, vol. 115, p. 102016, 2021.

[193] H. Gupta, A. Vahid Dastjerdi, S. K. Ghosh, and R. Buyya, "ifogsim: A toolkit for modeling and simulation of resource management techniques in the internet of things, edge and fog computing environments," *Software: Practice and Experience*, vol. 47, no. 9, pp. 1275–1296, 2017.

[194] R. Mahmud, S. Pallewatta, M. Goudarzi, and R. Buyya, "Ifogsim2: An extended ifogsim simulator for mobility, clustering, and microservice management in edge and fog computing environments," *Journal of Systems and Software*, vol. 190, p. 111351, 2022.

[195] K. Ergun, X. Yu, N. Nagesh, L. Cherkasova, P. Mercati, R. Ayoub, and T. Rosing, "Reliot: Reliability simulator for iot networks," in *International Conference on Internet of Things*. Springer, 2020, pp. 63–81.

[196] I. Lera, C. Guerrero, and C. Juiz, "Yafs: A simulator for iot scenarios in fog computing," *IEEE Access*, vol. 7, pp. 91 745–91 758, 2019.

[197] S. Tuli, S. R. Poojara, S. N. Srirama, G. Casale, and N. R. Jennings, "Cosco: Container orchestration using co-simulation and gradient based optimization for fog computing environments," *IEEE Transactions on Parallel and Distributed Systems*, vol. 33, no. 1, pp. 101–116, 2021.

[198] M. Scarpiniti, E. Baccarelli, A. Momenzadeh, and S. Sarv Ahrabi, "Deepfogsim: A toolbox for execution and performance evaluation of the inference phase of conditional deep neural networks with early exits atop distributed fog platforms," *Applied Sciences*, vol. 11, no. 1, p. 377, 2021.

[199] T. Qayyum, A. W. Malik, M. A. K. Khattak, O. Khalid, and S. U. Khan, "Fognetsim++: A toolkit for modeling and simulation of distributed fog environment," *IEEE Access*, vol. 6, pp. 63 570–63 583, 2018.

[200] R. N. Calheiros, R. Ranjan, A. Beloglazov, C. A. De Rose, and R. Buyya, "Cloudsim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms," *Software: Practice and experience*, vol. 41, no. 1, pp. 23–50, 2011.

[201] S. S. Gill, S. Tuli, A. N. Toosi, F. Cuadrado, P. Garraghan, R. Bahsoon, H. Lutfiyya, R. Sakellariou, O. Rana, S. Dustdar *et al.*, "Thermosim: Deep learning based framework for modeling and simulation of thermal-aware resource management for cloud computing environments," *Journal of Systems and Software*, vol. 166, p. 110596, 2020.

[202] C. Luo, F. Zhang, C. Huang, X. Xiong, J. Chen, L. Wang, W. Gao, H. Ye, T. Wu, R. Zhou *et al.*, "Aiot bench: towards comprehensive benchmarking mobile and embedded device intelligence," in *International Symposium on Benchmarking, Measuring and Optimization*. Springer, 2018, pp. 31–35.

[203] A. Shukla, S. Chaturvedi, and Y. Simmhan, "Riotbench: An iot benchmark for distributed stream processing systems," *Concurrency and Computation: Practice and Experience*, vol. 29, no. 21, p. e4257, 2017.

[204] T. Hao, Y. Huang, X. Wen, W. Gao, F. Zhang, C. Zheng, L. Wang, H. Ye, K. Hwang, Z. Ren *et al.*, "Edge aibench: towards comprehensive end-to-end edge computing benchmarking," in *International Symposium on Benchmarking, Measuring and Optimization*. Springer, 2018, pp. 23–30.

[205] S. Shen, V. Van Beek, and A. Iosup, "Statistical characterization of business-critical workloads hosted in cloud datacenters," in *2015 15th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing*. IEEE, 2015, pp. 465–474.

[206] E. Cortez, A. Bonde, A. Muzio, M. Russinovich, M. Fontoura, and R. Bianchini, "Resource central: Understanding and predicting workloads for improved resource management in large cloud platforms," in *Proceedings of the 26th Symposium on Operating Systems Principles*, 2017, pp. 153–167.

[207] M. Merenda, C. Porcaro, and D. Iero, "Edge machine learning for ai-enabled iot devices: A review," *Sensors*, vol. 20, no. 9, p. 2533, 2020.

[208] M. S. Murshed, C. Murphy, D. Hou, N. Khan, G. Ananthanarayanan, and F. Hussain, "Machine learning at the network edge: A survey," *ACM Computing Surveys (CSUR)*, vol. 54, no. 8, pp. 1–37, 2021.

[209] Y.-L. Lee, P.-K. Tsung, and M. Wu, "Techology trend of edge ai," in *2018 International Symposium on VLSI Design, Automation and Test (VLSI-DAT)*. IEEE, 2018, pp. 1–2.

[210] C. Liu, C. Liu, Y. Shang, S. Chen, B. Cheng, and J. Chen, "An adaptive prediction approach based on workload pattern discrimination in the cloud," *Journal of Network and Computer Applications*, vol. 80, pp. 35–44, 2017.

[211] A. A. Bankole and S. A. Ajila, "Predicting cloud resource provisioning using machine learning techniques," in *2016 26th IEEE Canadian Conference on Electrical and Computer Engineering (CCECE)*. IEEE, 2016, pp. 1–4.

[212] D. Didona, F. Quaglia, P. Romano, and E. Torre, "Enhancing performance prediction robustness by combining analytical modeling and machine learning," in *Proceedings of the 6th ACM/SPEC international conference on performance engineering*, 2015, pp. 145–156.

[213] R. Marcus and O. Papaemmanouil, "Workload management for cloud databases via machine learning," in *2016 IEEE 32nd International Conference on Data Engineering Workshops (ICDEW)*. IEEE, 2016, pp. 27–30.

[214] F.-J. Ferrández-Pastor, H. Mora, A. Jimeno-Morenilla, and B. Volckaert, "Deployment of iot edge and fog computing technologies to develop smart building services," *Sustainability*, vol. 10, no. 11, p. 3832, 2018.

[215] J. He, J. Wei, K. Chen, Z. Tang, Y. Zhou, and Y. Zhang, "Multitier fog computing with large-scale iot data analytics for smart cities," *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 677–686, 2017.

[216] L. Shooshtarian, D. Lan, and A. Taherkordi, "A clustering-based approach to efficient resource allocation in fog computing," in *International Symposium on Pervasive Systems, Algorithms and Networks*. Springer, 2019, pp. 207–224.

[217] Y. Meng, R. Rao, X. Zhang, and P. Hong, "Crupa: A container resource utilization prediction algorithm for auto-scaling based on time series analysis," in *2016 International conference on progress in informatics and computing (PIC)*. IEEE, 2016, pp. 468–472.

[218] S. Venkateswaran and S. Sarkar, "Fitness-aware containerization service leveraging machine learning," *IEEE Transactions on Services Computing*, vol. 14, no. 6, pp. 1751–1764, 2019.

[219] R. Yang, X. Ouyang, Y. Chen, P. Townend, and J. Xu, "Intelligent resource scheduling at scale: a machine learning perspective," in *2018 IEEE symposium on service-oriented system engineering (SOSE)*. IEEE, 2018, pp. 132–141.

[220] V. Podolskiy, M. Mayo, A. Koay, M. Gerndt, and P. Patros, "Maintaining slos of cloud-native applications via self-adaptive resource sharing," in *2019 IEEE 13th International Conference on Self-Adaptive and Self-Organizing Systems (SASO)*. IEEE, 2019, pp. 72–81.

[221] A. Mozo, B. Ordozgoiti, and S. Gómez-Canaval, "Forecasting short-term data center network traffic load with convolutional neural networks," *PLOS one*, vol. 13, no. 2, p. e0191939, 2018.

[222] J. Kumar, R. Goomer, and A. K. Singh, "Long short term memory recurrent neural network (lstm-rnn) based workload forecasting model for cloud datacenters," *Procedia Computer Science*, vol. 125, pp. 676–682, 2018.

[223] Q. Zhang, L. T. Yang, Z. Yan, Z. Chen, and P. Li, "An efficient deep learning model to predict cloud workload for industry informatics," *IEEE transactions on industrial informatics*, vol. 14, no. 7, pp. 3170–3178, 2018.

[224] S. Jeddi and S. Sharifian, "A water cycle optimized wavelet neural network algorithm for demand prediction in cloud computing," *Cluster Computing*, vol. 22, no. 4, pp. 1397–1412, 2019.

[225] D. Bega, M. Gramaglia, M. Fiore, A. Banchs, and X. Costa-Perez, "Deepcog: Optimizing resource provisioning in network slicing with ai-based capacity forecasting," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 2, pp. 361–376, 2019.

[226] S. Levy, R. Yao, Y. Wu, Y. Dang, P. Huang, Z. Mu, P. Zhao, T. Ramani, N. Govindaraju, X. Li *et al.*, "Predictive and adaptive failure mitigation to avert production cloud {VM} interruptions," in *14th USENIX Symposium on Operating Systems Design and Implementation (OSDI 20)*, 2020, pp. 1155–1170.

[227] H. Feng, Y. Jiang, D. Niyato, F.-C. Zheng, and X. You, "Content popularity prediction via deep learning in cache-enabled fog radio access networks," in *2019 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2019, pp. 1–6.

[228] W. Zhang, P. Duan, L. T. Yang, F. Xia, Z. Li, Q. Lu, W. Gong, and S. Yang, "Resource requests prediction in the cloud computing environment with a deep belief network," *Software: Practice and Experience*, vol. 47, no. 3, pp. 473–488, 2017.

[229] P. Yazdanian and S. Sharifian, "E2lg: a multiscale ensemble of lstm/gan deep learning architecture for multistep-ahead cloud workload prediction," *The Journal of Supercomputing*, vol. 77, no. 10, pp. 11 052–11 082, 2021.

[230] S.-s. Lee and S. Lee, "Resource allocation for vehicular fog computing using reinforcement learning combined with heuristic information," *IEEE Internet of Things Journal*, vol. 7, no. 10, pp. 10 450–10 464, 2020.

[231] S. Ouhame, Y. Hadi, and A. Ullah, "An efficient forecasting approach for resource utilization in cloud data center using cnn-lstm model," *Neural Computing and Applications*, vol. 33, no. 16, pp. 10 043–10 055, 2021.

[232] F. Fu, Y. Kang, Z. Zhang, F. R. Yu, and T. Wu, "Soft actor–critic drl for live transcoding and streaming in vehicular fog-computing-enabled iov," *IEEE Internet of Things Journal*, vol. 8, no. 3, pp. 1308–1321, 2020.

[233] X. Zhang, Y. Xiao, Q. Li, and W. Saad, "Deep reinforcement learning for fog computing-based vehicular system with multi-operator support," in *ICC 2020-2020 IEEE International Conference on Communications (ICC)*. IEEE, 2020, pp. 1–6.

[234] N. Van Huynh, D. T. Hoang, D. N. Nguyen, and E. Dutkiewicz, "Optimal and fast real-time resource slicing with deep dueling neural networks," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 6, pp. 1455–1470, 2019.

[235] Z. Cheng, M. Min, M. Liwang, L. Huang, and Z. Gao, "Multiagent ddpg-based joint task partitioning and power control in fog computing networks," *IEEE Internet of Things Journal*, vol. 9, no. 1, pp. 104–116, 2021.

[236] H. Sami, H. Otrok, J. Bentahar, and A. Mourad, "Ai-based resource provisioning of ioe services in 6g: A deep reinforcement learning approach," *IEEE Transactions on Network and Service Management*, vol. 18, no. 3, pp. 3527–3540, 2021.

[237] Z. Xu, J. Tang, C. Yin, Y. Wang, G. Xue, J. Wang, and M. C. Gursoy, "Recarl: resource allocation in cloud rans with deep reinforcement learning," *IEEE Transactions on Mobile Computing*, 2020.

[238] M. Chen, T. Wang, S. Zhang, and A. Liu, "Deep reinforcement learning for computation offloading in mobile edge computing environment," *Computer Communications*, vol. 175, pp. 1–12, 2021.

[239] S. Tuli, S. Ilager, K. Ramamohanarao, and R. Buyya, "Dynamic scheduling for stochastic edge-cloud computing environments using a3c learning and residual recurrent neural networks," *IEEE Transactions on Mobile Computing*, 2020.

[240] Y. Hu, C. de Laat, and Z. Zhao, "Learning workflow scheduling on multi-resource clusters," in *2019 IEEE International Conference on Networking, Architecture and Storage (NAS)*. IEEE, 2019, pp. 1–8.

[241] G. R. Ghosal, D. Ghosal, A. Sim, A. V. Thakur, and K. Wu, "A deep deterministic policy gradient based network scheduler for deadline-driven data transfers," in *2020 IFIP Networking Conference (Networking)*. IEEE, 2020, pp. 253–261.

[242] S. Sheng, P. Chen, Z. Chen, L. Wu, and Y. Yao, "Deep reinforcement learning-based task scheduling in iot edge computing," *Sensors*, vol. 21, no. 5, p. 1666, 2021.

[243] S. Bian, X. Huang, and Z. Shao, "Online task scheduling for fog computing with multi-resource fairness," in *2019 IEEE 90th Vehicular Technology Conference (VTC2019-Fall)*. IEEE, 2019, pp. 1–5.

[244] D. Van Le and C.-K. Tham, "A deep reinforcement learning based offloading scheme in ad-hoc mobile clouds," in *IEEE INFOCOM 2018-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, 2018, pp. 760–765.

[245] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski *et al.*, "Human-level control through deep reinforcement learning," *nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[246] B. Hazarika, K. Singh, S. Biswas, and C.-P. Li, "Drl-based resource allocation for computation offloading in iov networks," *IEEE Transactions on Industrial Informatics*, 2022.

[247] J. Cen and Y. Li, "Resource allocation strategy using deep reinforcement learning in cloud-edge collaborative computing environment," *Mobile Information Systems*, vol. 2022, 2022.

[248] H. Wu, Z. Zhang, C. Guan, K. Wolter, and M. Xu, "Collaborate edge and cloud computing with distributed deep learning for smart city internet of things," *IEEE Internet of Things Journal*, vol. 7, no. 9, pp. 8099–8110, 2020.

[249] H. Rafique, M. A. Shah, S. U. Islam, T. Maqsood, S. Khan, and C. Maple, "A novel bio-inspired hybrid algorithm (nbiha) for efficient resource management in fog computing," *IEEE Access*, vol. 7, pp. 115 760–115 773, 2019.
[250] A. M. Abdelmoniem and M. Canini, "Dc2: Delay-aware compression control for distributed machine learning," in *IEEE INFOCOM 2021-IEEE Conference on Computer Communications*. IEEE, 2021, pp. 1–10.
[251] H. Wu, K. Wolter, P. Jiao, Y. Deng, Y. Zhao, and M. Xu, "Eedto: an energy-efficient dynamic task offloading algorithm for blockchain-enabled iot-edge-cloud orchestrated computing," *IEEE Internet of Things Journal*, vol. 8, no. 4, pp. 2163–2176, 2020.
[252] M. Xue, H. Wu, R. Li, M. Xu, and P. Jiao, "Eosdnn: An efficient offloading scheme for dnn inference acceleration in local-edge-cloud collaborative environments," *IEEE Transactions on Green Communications and Networking*, vol. 6, no. 1, pp. 248–264, 2021.

Authors Biography

Sundas Iftikhar is a Ph.D. Scholar at the School of Electronic Engineering and Computer Science, Queen Mary University of London. Prior to this, she held positions as Research associate and Lecturer at University of Kotli Azad Jammu and Kashmir, Azad Kashmir, Pakistan. She did her Masters in computer software engineering from National University of Science and Technology, Pakistan. Her research interest include Cloud computing, Fog computing, and resource Management in Fog.

Sukhpal Singh Gill is a Lecturer (Assistant Professor) in Cloud Computing at the School of Electronic Engineering and Computer Science, Queen Mary University of London, UK. Prior to his present stint, Dr. Gill has held positions as a Research Associate at the School of Computing and Communications, Lancaster University, UK and also as a Postdoctoral Research Fellow at CLOUDS Laboratory, The University of Melbourne, Australia. Dr. Gill is serving as an Associate Editor in Wiley ETT and IET Networks Journal. He has co-authored 70+ peer-reviewed papers (with H-index 30+) and has published in prominent international journals and conferences such as IEEE TCC, IEEE TSC, IEEE TII, IEEE TNSM, IEEE IoT Journal, Elsevier JSS/FGCS, IEEE/ACM UCC and IEEE CCGRID. He has received several awards, including the Distinguished Reviewer Award from SPE (Wiley), 2018, Best Paper Award AusPDC at ACSW 2021 and has also served as the PC member for venues such as PerCom, UCC, CCGRID, CLOUDS, ICFEC, AusPDC. His research interests include Cloud Computing, Fog Computing, Software Engineering, Internet of Things and Energy Efficiency. For further information, please visit http://www.ssgill.me.

Chenghao Song received his BSc degree from University of Electronic Science and Technology of China. Now he is a master student at University of Melbourne, he is also a visiting student at Shenzhen Institutes of Advanced Technology, Chinese Academy of Science. His research interest includes deep learning for cloud resource optimization.

Minxian Xu is currently an associate professor at Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences. He received the B.Sc. degree and the M.Sc. degree, both in software engineering from University of Electronic Science and Technology of China. He obtained his Ph.D. degree from the University of Melbourne in 2019. His research interests include resource scheduling and optimization in cloud computing. He has co-authored 30+ peer-reviewed papers published in prominent international journals and conferences, such as ACM Computing Surveys, IEEE Transactions on Sustainable Computing, IEEE Transactions on Cloud Computing, Journal of Parallel and Distributed Computing, Software: Practice and Experience, International Conference on Service-Oriented Computing. His Ph.D. Thesis was awarded the 2019 IEEE TCSC Outstanding Ph.D. Dissertation Award. He is member of CCF and IEEE. More information can be found at: http://www.minxianxu.info.

Mohammad Sadegh Aslanpour is a PhD student in Monash University and CSIRO's DATA61, Australia. He obtained his MSc degree in Computer Engineering Islamic Azad University, Tehran Science and Research (Sirjan) Branch, Iran in 2016. He also obtained his Bachelor and Associate degrees in Computer-Software in 2012 and 2010, respectively. From 2011 to 2019, he worked in the industry, IT Department of Jahrom Municipality, Iran as Software Engineer. He is also serving as Editorial Board Member and Reviewer for some international high-ranked journals. His research interests include orchestration of Cloud, Fog and Edge Computing, Serverless Computing, and Autonomous

Systems. For more details, please visit his homepage: aslanpour.github.io

Adel N. Toosi is a senior lecturer (a.k.a. Associate Professor) at Department of Software Systems and Cybersecurity, Faculty of Information Technology, Monash University, Australia. Before joining Monash, Dr Toosi was a Postdoctoral Research Fellow at the University of Melbourne from 2015 to 2018. He received his Ph.D. degree in 2015 from the School of Computing and Information Systems at the University of Melbourne. His Ph.D. thesis was nominated for CORE John Makepeace Bennett Award for the Australasian Distinguished Doctoral Dissertation and John Melvin Memorial Scholarship for the Best Ph.D. thesis in Engineering. Dr Toosi made significant contributions to the areas of resource management and software systems for cloud computing. His research interests include Cloud/Fog/Edge Computing, Software Defined Networking, Green Computing and Energy Efficiency. Currently, he is working on green energy harvesting for Edge/Fog computing environments. For further information, please visit his homepage: http://adelndjarantoosi.info

Junhui Du received the BSc degree in mathematics from the Nanjing University of Information Science Technology, China, in 2021. He is currently working toward the master's degree in mathematics with the Center for Applied Mathematics, Tianjin University, China. His research interests include Internet of Things, deep learning, and mobile edge computing.

Huaming Wu received the BE and MS degrees from the Harbin Institute of Technology, China, in 2009 and 2011, respectively, both in electrical engineering, and the PhD degree with the highest honor in computer science at Freie Universität Berlin, Germany, in 2015. He is currently an associate professor with the Center for Applied Mathematics, Tianjin University, China. His research interests include model-based evaluation, wireless and mobile network systems, mobile cloud computing and deep learning. He is a senior member of IEEE and a member of ACM. For further information, please visit: http://huamingwu.cn.

Shreya Ghosh is a Postdoctoral Research Fellow at The Pennsylvania State University, Pennsylvania, USA. She received her PhD from the Department of Computer Science and Engineering, IIT Kharagpur, India in 2021. Her current research interests include machine learning, trajectory data mining, cloud computing and Internet of Spatial Things. Shreya is the recipient of the prestigious TCS fellowship. She is the member of AnitaB.org and member of IEEE and ACM.

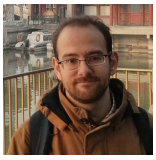
Deepraj Chowdhury is with department of electronics and communication, IIIT- Naya Raipur. He has Co-authored 4 research papers in different conferences like ICACCP 2021, INDICON 2021. He also has 3 Indian copyright registered, and applied for 2 Indian Patent. He is also serving as a reviewer in Wiley Transaction on emerging Telecommunication Technologies.

Muhammed Golec is a PhD student in Computer Science at Queen Mary University. After his undergraduate graduation, he was awarded the Ministry of Education Scholarship, one of the most prestigious scholarships in his country. Within the scope of this scholarship, he graduated from Queen Mary University of London Computer Science with a high degree (Distinction). His master thesis was found successful and published in IEEE Consumer Electronics Magazine. He worked at Sisecam Company as an Electrical and Electronics Maintenance Engineer for one year to consolidate his academic skills in the private sector. His research interests include AI, Cloud Computing, and Security and Privacy. For further information, please visit https://www.linkedin.com/in/muhammed-golec-b55756119/.

Mohit Kumar is Assistant Professor in the Department of Information Technology at Dr. B R Ambedkar National Institute of Technology, Jalandhar, India. He received his Ph.D. degree from Indian Institute of Technology Roorkee in the field of Cloud Computing, 2018, and M.Tech degree in Computer Science and Engineering from ABV-Indian Institute of Information Technology Gwalior, India in 2013. He has received his B.Tech degree in Computer Science and Engineering from MJP Rohilkhand University Bareilly, 2009. His research topics cover the areas of Cloud computing, Fog computing, Edge Computing, Internet of Things, Soft Computing. He has published more than 20 research articles in reputed journals and international conferences. He has been Session chair and keynotes Speaker of many International

conferences, webinars, FDP, STC in India. He has guided two M.Tech Thesis and guided 1 Ph.D. Scholar. He is an active reviewer of several reputed journals and international conferences. He is a member of the IEEE.

Ahmed M. Abdelmoniem is a Lecturer (Assistant Professor) of Big Data and Distributed Systems at the School of EECS, QMUL and leads the Scalable Adaptive Yet Efficient Distributed (SAYED) Systems Research Group. He has a PhD in Computer Science and Engineering from the Hong Kong University of Science and Technology (HKUST), Hong Kong. His research interests lie in the intersection of distributed systems, computer networks and machine learning. He is an investigator on several UK and international grants totalling nearly USD 650K in funding. His work appears in top-tier conferences and journals including NeurIPS, AAAI, MLSys, ACM EuroSys, IEEE INFOCOM, IEEE ICDCS, and IEEE/ACM Transactions on Networking. He is interested in supervising students with a background in Computer Networks, Machine Learning, Distributed Systems and Cloud/Edge Computing



Felix Cuadrado received the Ph.D. degree in telecommunications engineering from the Universidad Politécnica de Madrid (UPM), Spain, in 2009. He is currently a Senior Distinguished Fellow (Beatriz Galindo scheme) with the Universidad Politécnica de Madrid, a Visiting Reader at the Queen Mary University of London, and a fellow of the Alan Turing Institute. He has numerous publications in top tier journals and conferences, including IEEE TSC, IEEE TCC, Elsevier JSS, Elsevier FCGS, IEEE ICDCS, and WWW. His research explores the challenges arising from large-scale data-intensive applications through a combination of software engineering, distributed systems, and mathematical approaches

Blesson Varghese received the Ph.D. degree in computer science from the University of Reading, UK on international scholarships. He is a Reader (Associate Professor) in computer science at the University of St Andrews and an honorary faculty member at Queen's University Belfast. He is the Principal Investigator of the Edge Computing Hub and was a Royal Society Short Industry Fellow to British Telecommunications plc. His interests include developing and analysing novel parallel and distributed systems and applications that span the cloud–edge–device continuum. More information is available from http://www.blessonv.com

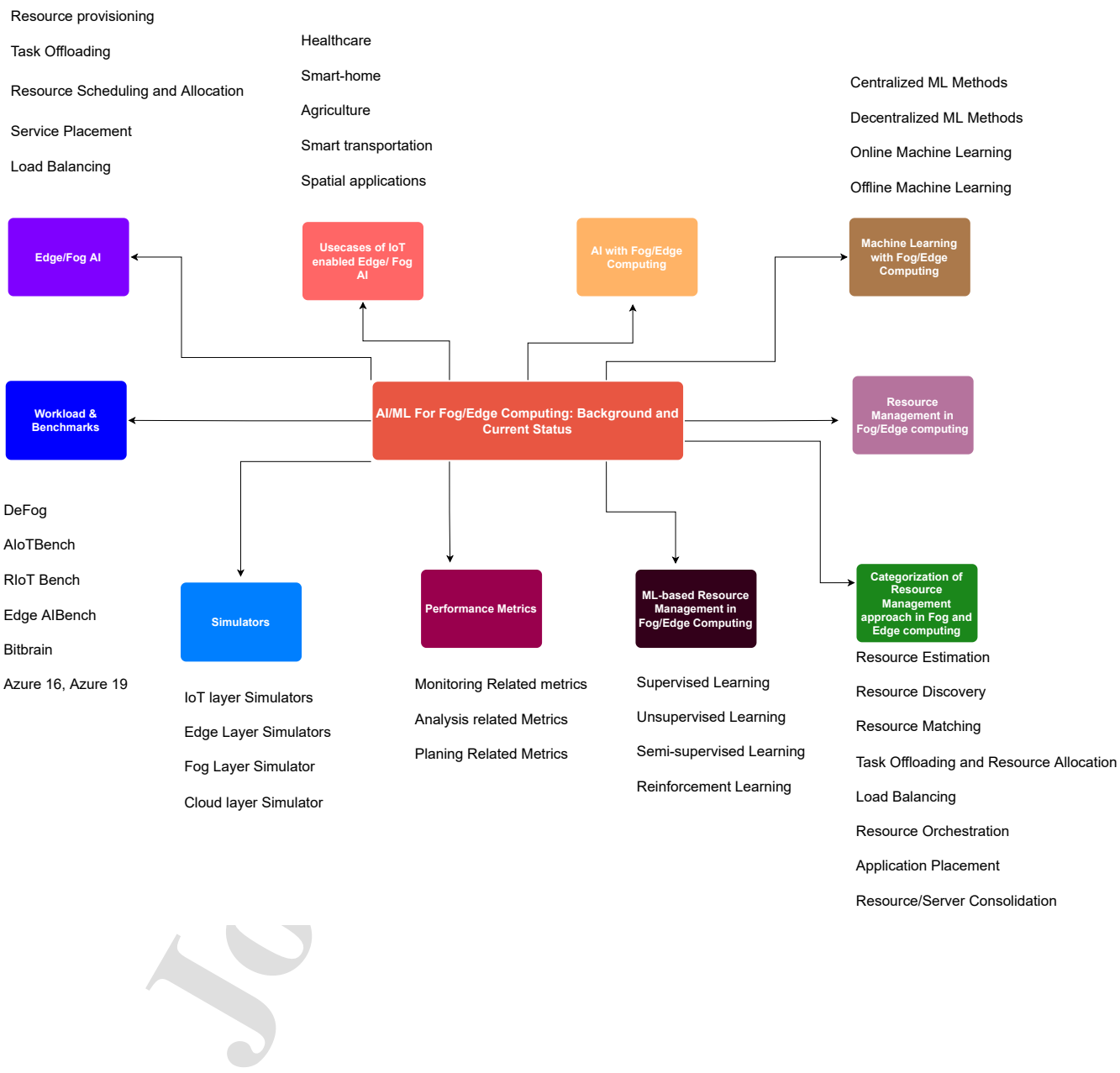
Omer Rana is a Professor of Performance Engineering in School of Computer Science & Informatics at Cardiff University and Deputy Director of the Welsh e-Science Centre. He holds a Ph.D. from Imperial College. His research interests extend to three main areas within computer science: problem solving environments, high performance agent systems and novel algorithms for data analysis and management. Moreover, he leads the Complex Systems research group in the School of Computer Science & Informatics and is director of the 'Internet of Things' Lab, at Cardiff University. He has published over 310 papers in peer-reviewed international conferences and journals. He serves on the Editorial Board of IEEE Transactions on Parallel and Distributed Systems, ACM Transactions on Internet Technology, and ACM Transactions on Autonomous and Adaptive Systems. He has served as a Co-Editor for a number of journals, including Concurrency: Practice and Experience (John Wiley), IEEE Transactions on System, Man, and Cybernetics: Systems, and IEEE Transactions on Cloud Computing.

Schahram Dustdar is Full Professor of computer science heading the Research Division of Distributed Systems at the TU Wien, Austria. He is founding Co-Editor-in-Chief of the new ACM Transactions on Internet of Things (ACM TIoT) as well as Editor-in-Chief of Computing (Springer). He is an Associate Editor of IEEE Transactions on Services Computing, IEEE Transactions on Cloud Computing, ACM Transactions on the Web, and ACM Transactions on Internet Technology, as well as on the editorial board of IEEE Internet Computing and IEEE Computer. Dustdar is IEEE Fellow (2016), recipient of the ACM Distinguished Scientist Award (2009), the ACM Distinguished Speaker ward (2021), the IBM Faculty Award (2012), an Elected Member of the Academia Europaea: The Academy of Europe, where he is Chairman of the Informatics Section. In 2021 Dustdar was elected EAI Fellow as well as Fellow and President for the Asia-Pacific Artificial Intelligence Association (AAIA).

Steve Uhlig obtained a Ph.D. degree in Applied Sciences from the University of Louvain, Belgium, in 2004. From 2004 to 2006, he was a Postdoctoral Fellow of the Belgian National Fund for Scientific Research (F.N.R.S.). His thesis won the annual IBM Belgium/F.N.R.S. Computer Science Prize 2005. Between 2004 and 2006, he was a visiting scientist at Intel Research Cambridge, UK, and at the Applied Mathematics Department of University of Adelaide, Australia. Between 2006 and 2008, he was with Delft University of Technology, the Netherlands. Prior to joining Queen Mary University of London, he was a Senior Research Scientist with Technische Universität Berlin/Deutsche Telekom Laboratories, Berlin, Germany. Since January 2012, he has been the Professor of Networks and Head of the Networks Research group at Queen Mary, University of London. Between 2012 and 2016, he was a guest professor at the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China. With expertise in network monitoring, large-scale network measurements and analysis, and network engineering, during his career he has been published in over 100 peer-reviewed journals, and awarded over £3million in grant funding. Awarded a Turing Fellow, Steve is also the Principal Investigator on a new project funded by the Alan Turing Institute: 'Learning-based reactive Internet Engineering' (LIME). He is currently the Editor in Chief of ACM SIGCOMM Computer Communication Review, the newsletter of the ACM SIGCOMM SIG on data communications. Since December 2020, Steve has also held the position of Head of School of Electronic Engineering and Computer Science. Current Research interests: Internet measurements, software-defined networking, content delivery.

Highlights

- Review AI/ML approaches used for the realization of AI/ML for fog/edge computing.
- Discuss the background of AI/ML for fog/edge computing with broad taxonomy.
- Present a current status of AI/ML-based resource management in fog/edge computing.
- Propose a taxonomy of AI/ML-based resource management methods in fog/edge computing.
- Identify open issues and future directions for the union of edge and AI as Edge AI.



Declaration of interests

☒ The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐ The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: