Chapter 6
Machine Learning for IoT Systems

Ahmed Khattab and Nouran Youssry

Abstract The rapid increase in the number of smart devices hosting sophisticated applications is significantly affecting the landscape of the information communication technology industry. The Internet of Things (IoT) is gaining popularity and importance in man's everyday life. However, the IoT challenges also increase with its evolution. The urge for IoT improvement and continuous enhancement becomes more important. Machine learning techniques are recently being exploit-ed within IoT systems to leverage their potential. This chapter comprehensively surveys of the use of algorithms that exploit machine learning in IoT systems. We classify such machine learning-based IoT algorithms into those which provide ef-ficient solutions to the IoT basic operation challenges, such as localization, clus-tering, routing and data aggregation, and those which target performance-related challenges, such as congestion control, fault detection, resource management and security.

Keywords Internet of Things (IoT) · Wireless sensor network (WSN) · Machine learning · Unsupervised learning · Supervised learning · Fuzzy logic

6.1 Introduction

The Internet of Things (IoT) is a networking paradigm that offers pervasive and distributed services in the move towards ubiquitous computing. IoT is a network of objects or things that communicate with each other and with the surrounding environment and share information through the Internet. IoT enables millions of devices, including sensors and smart phones/devices, to be connected for performing different tasks. According to the International Data Cooperation (IDC), the number of

A. Khattab (✉) · N. Youssry
Electronics and Electrical Communications Engineering Department, Cairo University, Giza, Egypt
e-mail: akhattab@ieee.org

M. Alam et al. (eds.), *Internet of Things (IoT)*, S.M.A.R.T. Environments,
https://doi.org/10.1007/978-3-030-37468-6_6

IoT devices worldwide will exceed 50 billion by 2020 producing more than 60 ZB of data (Van der 2017; Sam 2016).

Wireless Sensor Networks (WSNs) are playing a main role in IoT. WSNs have attracted significant attention in recent years. A WSN is composed of a set of application-specific sensor nodes equipped with communication modules. Such nodes gather data from their environment to monitor and record target conditions at diverse locations. While there are sensors that measure almost every environmental aspect, the widely monitored parameters are air temperature and humidity, wind speed and direction, illumination intensity, flow pressure, vibration intensity, sound intensity, power-line voltage, pollution levels, chemical concentrations, and vital body functions. WSNs are powerful in developing application-specific systems.

WSNs joins the IoT networking paradigm when the sensor nodes dynamically connect to the Internet to cooperate to achieve their tasks. Both IoT and WSNs face several challenges and issues that should be addressed. Examples include energy efficiency, node localization and clustering, event scheduling, route establishment, data aggregation, fault detection and data security. Exploiting machine learning provides solutions to such problems. Machine learning could significantly boost the performance and distributive characteristic of IoT.

Machine learning (ML) emerged as an artificial intelligence (AI) technique in the late 1950s (Ayodele 2010). Since then, its algorithms gradually evolved to become more robust, effective and accurate. Recently, ML classification and regressing techniques have been widely exploited to improve the performance of many of application domains such as bioinformatics, facial and speech recognition, agriculture monitoring, fraud detection and marketing.

Machine learning could be used to improve the performance of various IoT systems by exploiting the history of the collected data of given tasks to autonomously optimize the performance without the need to re-program the system. More specifically, the main reasons that make ML important in IoT applications are:

- The rapidly changing dynamic nature of the environments typically monitored by IoT systems. Therefore, developing IoT systems that efficiently operate by autonomously adapting to such changes is required.
- The unreachable and dangerous settings in which exploratory IoT applications, such as wastewater and volcano eruption monitoring, operate to collect new knowledge. Consequently, the ability of ML-based IoT systems to self-calibrate to the acquired new knowledge is needed to ensure robustness.
- Machine learning does not only improve the autonomous control in IoT applications but also ameliorate their intelligent decision-making capabilities.

Nevertheless, the use of ML in IoT still face several challenges that should be carefully considered. For instance, IoT devices are resource limited. Using ML to extract consensus relationships between the collected data samples and predicting the accurate hypotheses significantly drain the energy of the IoT devices. This necessitates trading-off the ML algorithm's computational complexity and the targeted accuracy of the learning process.

In this chapter, we present a brief introduction of IoT: its concept, history, architecture and its processing data layers. We also present a comprehensive survey of machine learning techniques classifying them into five categories which are supervised learning, unsupervised learning, reinforcement learning, evolutionary computational and fuzzy logic techniques. We present a detailed study of the applications of machine learning techniques in solving IoT challenges. Moreover, we classify those applications into operational applications which are concerned with the main functions of the IoT system and performance applications which are more concerned with enhancing the performance of IoT systems.

The remainder of the chapter is organized as follows. Section 6.2 briefly introduces IoT. Section 6.3 overviews of the different machine learning algorithms. The role of machine learning in solving operational and performance challenges in IoT and WSN systems are discussed in Sects. 6.4 and 6.5, respectively. Finally, our conclusions are drawn in Sect. 6.6.

6.2 IoT Overview

IoT is a network of objects "things" which sense, accumulate and transfer data over the Internet without any human intervention. Kevin Ashton, British technology pioneer and co-founder of MIT's Auto-ID Center, first used the term "Internet of Things" in 1999. Ashton used the term to illustrate the power of using Radio-Frequency Identification (RFID) tags to connect goods to the Internet, then count and track them without needing human intervention. Since then, the idea of globally connecting computers and servers through the Internet has been expanded to the Internet of Things in which anything can be connected and accessed through the Internet. This created a whole new connectivity dimension where anything can be connected at anytime and anyplace (Vashi et al. 2017). The industries that are adopting IoT are expected to achieve revenue growth of 22% (Kotha and Gupta 2018).

Several IoT architectures have been developed to handle the use of heterogeneous devices in such systems. The number and type of used devices, the application and the amount of collected and processed data control the choice of the most suitable architecture to be used.

One simple IoT architectural model is the three-layer model shown in Fig. 6.1 which consists of perception, network and application layers (Ghasempour 2019).

- Perception Layer: This is the physical layer of the IoT system. It is composed of the sensors which gather information about the environment and actuators which implement the actions that accordingly change the environment. A temperature controller in an air conditioner is an example of an actuator.
- Network layer: This is the transmission layer that is responsible for handling the routing decisions. It is also handling the transmission and processing of the data received from or transmitted to the perception layer.

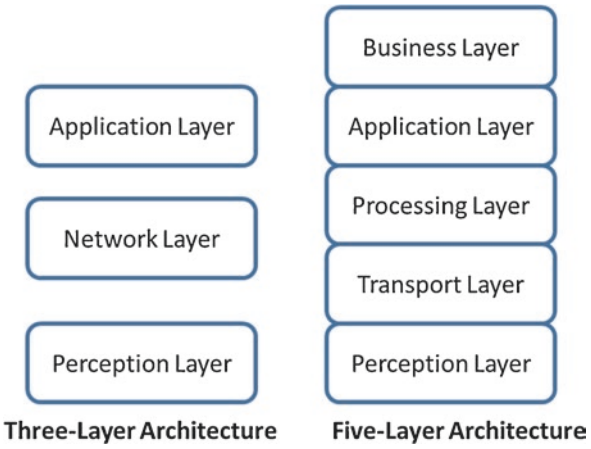

Fig. 6.1 IoT architecture models

- Application Layer: This layer delivers application-specific services to end users. It also provides the interface between humans and the IoT system.

Such three-layer architecture represents the simplest IoT architecture. As the data size of the system increase, this architecture becomes inefficient. That is why the five-layer model shown in Fig. 6.1 was proposed in (Sethi and Sarangi 2017), adding the following three layers to the perception and application layers:

- Transport Layer: This layer handovers data from the perception layer to the processing layer and actions in the reverse direction. Several network types are used for this purpose such as wireless LAN, NB-IoT, LoRA, RFID, and NFC.
- Processing Layer: This is the middleware layer that stores and processes the huge amounts of data received from the transport layer. It also prepares the data for the application layer. The processing layer manages and provides a wide range of services to the lower layers. Cloud computing, databases and big data analysis are examples of the technologies used in this layer.
- Business Layer: This layer encompasses the overall IoT application alongside its business and profit models. It is also responsible for the end users' privacy and security.

Another architecture presented in (Navani et al. 2017) proposed the same division of layers with only changing their names. The layers in this architecture are the object, object abstraction, service management, application and business layers.

Cloud computing was originally used to implement the processing layer because it provides significant flexibility and scalability. A cloud database management system based on a five-layer architecture was introduced in (Alam et al. 2013). Significant efforts were carried out to enhance cloud computing system's database with query processing mechanism in (Malhotra et al. 2018) and to enhance the task scheduler as in (Ali et al. 2019). As energy is known to be a scarce resource in IoT

systems, the authors of (Ali and Alam 2016) proposed energy management techniques for cloud computing environments. IoT devices generate valuable data readings that need to be transferred. Therefore, merging cloud computing technology with big data analysis (Alam and Shakil 2016) is a very important in many platforms as discussed in (Khan et al. 2016, 2018, 2019a, b, c, d; Shakil et al. 2017).

Lately, the increase of real-time applications requiring the least possible latency caused a migration towards another processing architectures that involve either fog or edge computing. In fog computing paradigms, the data and its processing take place in decentralized computing structures that physically reside between the data sources and the cloud. Hence, fog computing results in low-latency and is suitable for time-sensitive IoT applications as data is processed close to where it is originated. Its efficiency is also higher as less data is uploaded to the cloud. On the other hand, data processing takes place either on the IoT device generating the data or on a local gateway device that resides in the vicinity of the IoT device in the edge computing paradigm. Thus, both fog and edge computing reduce the dependence on the cloud infrastructure in data analysis, which in turn reduce the system latency, and hence, allow the data-driven decisions making process much faster. However, fog computing is the better option where data aggregation from different sources is needed whereas edge computing is better where the least latency is allowed. The differences between cloud computing, fog computing and edge computing are illustrated in Fig. 6.2.

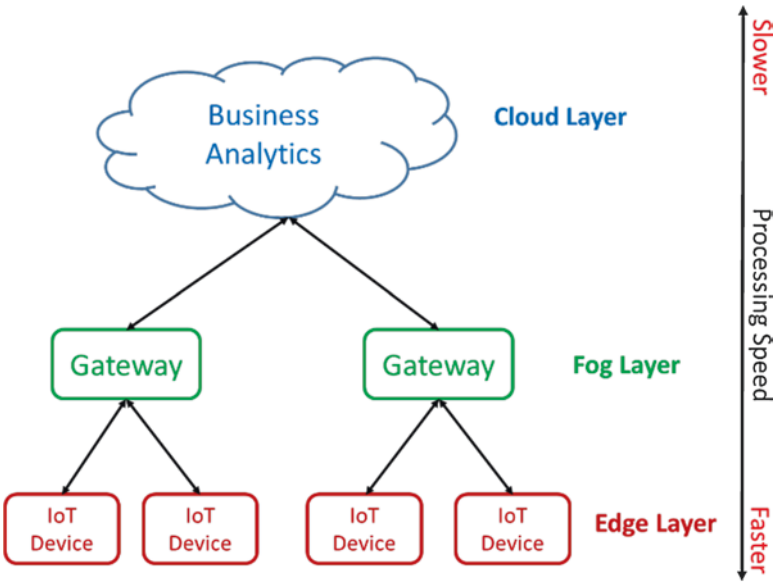

Fig. 6.2 Mapping IoT processing layers to system devices

6.3 Machine Learning Taxonomy

Machine learning techniques are designed to automatically benefit from prior experience in acting in the future without explicit reprogramming. Existing ML approaches are typically classified as either supervised, unsupervised or reinforcement learning. However, artificial intelligence techniques have recently played a great role in enhancing ML techniques. Therefore, this chapter categorizes ML techniques into supervised learning, unsupervised learning, reinforced learning, evolutionary computation and fuzzy logic. This section briefly overviews the different ML approaches in addition to their most updated algorithms in the context of IoT and WSNs.

6.3.1 Supervised Learning

In supervised learning, the input and targeted output data are both labeled for classification. This presents the learning base on which future data processing is centered. The key supervised learning algorithms are:

1. *k*-nearest Neighbor (*k*-NN): In this supervised learning approach, a data sample is classified according to the labels of nearby data samples. Simple methods (e.g., the Euclidean distance between the IoT devices) are typically used to compute the average measurements of neighboring devices within a specific range. It is a simple computational algorithm but may be inaccurate in large data sets. In IoT, *k*-NN is used in fault detection (Warriach and Tei 2017) and data aggregation approaches (Li and Parker 2014).
2. Support Vector Machine (SVM): Decision planes are used in SVM approaches to define decision boundaries. A decision plane separates groups of objects each with different class memberships as depicted by the example shown in Fig. 6.3.

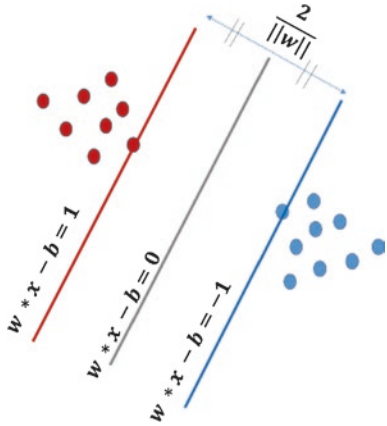

Fig. 6.3 Support vector machine example

SVM supervised learning is typically used for localization problems (Kang et al. 2018) to detect malicious behaviors and to address several security issues in IoT and WSNs (Zidi et al. 2018) due to its high accuracy.

3. Neural Network (NN): An artificial neural network (ANN) imitates biological neurons by interconnecting layers of artificial neurons. These artificial neurons map the different sets of input data onto a set of appropriate outputs. Figure 6.4 presents the model of a neural network. Even though ANNs provide solutions to non-linear and complex problems, they are computationally complex. Artificial NNs improve the efficiency of IoT localization (Banihashemian et al. 2018; El Assaf et al. 2016), detect faulty nodes (Chanak and Banerjee 2016), and establish routing (Mehmood et al. 2017).
4. Bayesian Interface: Unlike most machine learning algorithms, Bayesian inference uses a reasonably small number of samples for training. Bayesian methods efficiently learn uncertain perceptions by adapting probability distributions while avoiding overfitting. However, they need prior knowledge about the environment. Bayesian interface is suitable for fault detection (Warriach and Tei 2017), cluster head selection (Jafarizadeh et al. 2017) and localization (Sun et al. 2017; Wang et al. 2017; Guo et al. 2018) approaches.
5. Decision Tree (DT): A tree-like model of decision or classification is used in such a decision-support tool. DTs are created using a set of if-then conditions. For boosting DT accuracy, the random forest (RF) algorithm is introduced. RF is an ensemble decision tree method that operates by constructing multiple classifiers. Each classifier is a decision tree. RFs are used in intrusion detection as in (Varsha et al. 2017).

6.3.2 Unsupervised Learning

Unsupervised learning algorithms operate over datasets in which the input data does not have labeled responses. Inferences are drawn in such algorithms by classifying the unlabeled input data into groups that are called clusters.

1. Principal Component Analysis (PCA): PCA reduces the dimension of a large set of variables to a much smaller set that contains almost all the information in the

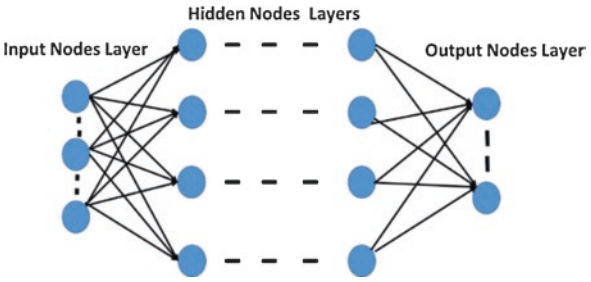

Fig. 6.4 Neural networks' structure

original large set. PCA application in IoT systems for data dimensionality reduction takes place either at the sensor or the cluster head levels. PCA results in a reduction in the communication overhead (Wang et al. 2019) which is very useful in data aggregation (Liu et al. 2017).

2. *k*-means Clustering: It is used to classify different data in classes or clusters (Jain et al. 2018). *k* random centroids are initially chosen. The other nodes then join the clusters of the nearest centroid. Averaging over the nodes in each cluster, new centroids are determined. The algorithm repeats the previous steps until convergence is reached.
3. Self-Organizing Maps (SOM): A self-organizing map is also considered as a method for dimensionality reduction as explained in (Miljković 2017). However, a SOM is a type of artificial neural network which result in a discretized low-dimension representation of the input data, called a map, using unsupervised learning for training. SOMs are very suitable to be used in building clusters in IoT.

6.3.3 Reinforcement Learning

Reinforcement learning (RL) does not have knowledge about the inputs nor their corresponding outputs. It is a very important ML technique whose idea, illustrated in Fig. 6.5, is that an agent will learn from the environment by interacting with it and receiving rewards for performing actions. Over the past few years, RL algorithms have been used for designing routing protocols in IoT systems and WSNs to reduce the energy consumption and improve the network performance (Habib et al. 2018).

An extensively used RL algorithm is Q-learning. First, a Q table is initialized, then an action a is performed. A reward is then measured to update the Q table. In order to assess how good to take a certain action a at a particular state s, the action-value function $Q(s, a)$ is learnt by the algorithm. Initially, the action a is randomly chosen until the Q table is constructed, then the best action is chosen from it.

Fig. 6.5 Reinforcement learning concept

6.3.4 *Evolutionary Computation*

Unlike other ML approaches, evolutionary computation techniques solve problems using computational models that mimic the biological behavior of either humans or animals in problem solving tasks.

1. Genetic Algorithms (GA): Such algorithms use biologically inspired heuristic search techniques to find the best solution for problems with large search spaces. GAs work in parallel on a population of solutions rather than processing a single solution. First, a chromosome structure is defined, typically in the form of an array of bits as shown in Fig. 6.6. Then, an initial chromosomes population is randomly generated for which fitness is evaluated. Chromosomes with higher fit-ness are selected in the selection process. A crossover process combines two parents to introduce a new child to the population. Finally, mutation randomly up-dates the parents to introduce new children. An example GA cycle is shown in Fig. 6.6. GAs are suitable for data aggregation approaches and searching for optimal clusters.
2. Ant Colony Optimization (ACO): ACO probabilistically searches for the optimal path in a graph in ways similar to how ants find the path between a food source and the colony. First, ants move randomly, leaving traces or pheromone on the taken path. More pheromone on a path indicates that the path probability to be the shortest/optimum one is high. This algorithm is efficient for routing in IoT.
3. Particle Swarm Optimization (PSO): PSO is inspired by swarm theory, fish schooling, and bird flocking. As an evolutionary computation approach, PSO searches for the best solution in a population. The algorithm starts with a random

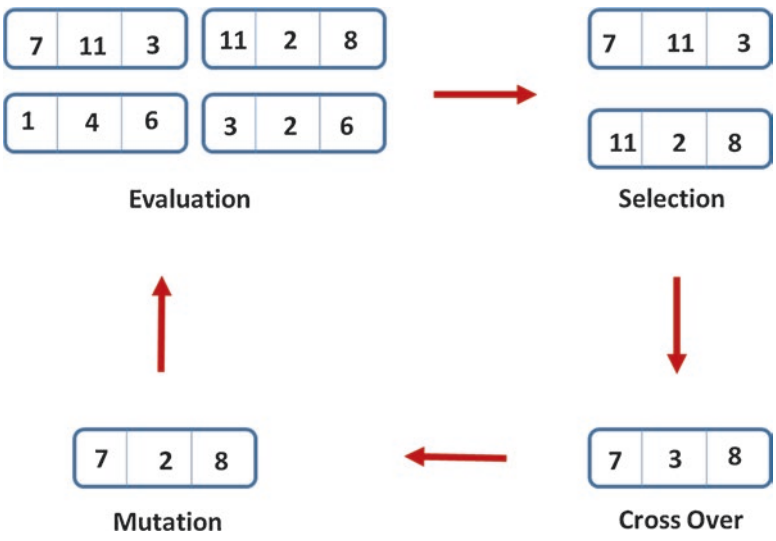

Fig. 6.6 Genetic algorithm example

population of solutions referred to as particles. A fitness function is used to compute the particle's fitness value which is optimized in each generation. IoT clustering algorithms recently exploit PSO to improve their performance.

6.3.5 Fuzzy Logic

Traditional ML techniques used to work with binary values: either 0 (False) and 1 (True). However, fuzzy logic (FL) imitates the way of decision making in a human which considers all the possibilities between 0 and 1 digital values (Umarikar 2003). FL introduces the concept of degree of truth. Its value does not have to be exactly 1. It can be any real value between 0 and 1 instead. Fuzzy logic is an attractive solution for localization typically used to combine the node's residual energy, centrality, and distance from the data sink node for electing the best cluster heads (Umarikar 2003).

6.4 Machine Learning for IoT Basic Operation

In this chapter, we categorize the challenges that face IoT systems into basic operation and performance-related challenges. In this section, we take a closer look on how ML is making an effective contribution in solving the basic system operation challenges such as node localization, clusters formation, routing, and data aggregation. Figure 6.7 summarizes how ML is used in addressing such challenges.

6.4.1 Node Localization

The procedure of determining the geographic coordinates of the nodes is known as localization. As much as it is crucial to be aware of IoT nodes' locations, it is impractical to use GPS hardware in every node as it dramatically consumes the nodes' energy. Alternatively, localization can exploit machine learning alongside some parameters such as the received signal strength (RSS) and the time and angle of arrival.

An approach was introduced in (Sun et al. 2018) to use neural networks in localizing WSN nodes. The proposed solution uses the variations of the RSS between the sensor nodes. RSS is measured from all the nodes once without the presence of any target, then with the target presence. The ANN uses the difference in RSS values and the corresponding matrix indices as inputs. The ANN outputs are the nodes' locations. The ANN is trained to approximate a nonlinear function to map the inputs and outputs.

The use of SVM in localization was proposed in (Kim et al. 2013). What makes this approach different is the use of ensemble SVM technique. The ensemble technique employs multiple ML techniques, then decides the result by voting. In

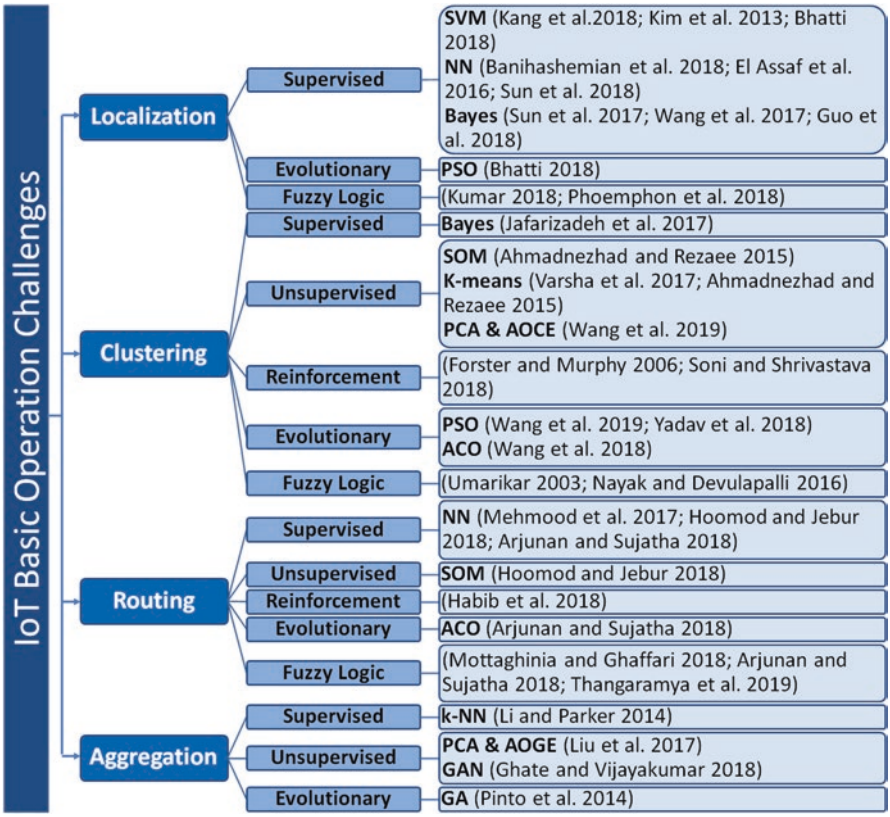

Fig. 6.7 Machine learning exploitation for IoT basic operation

(Kim et al. 2013), the authors used multiple SVMs by dividing the WSN to many subnetworks. Each subnetwork is trained using SVM. The resulting sub-predictions are then combined as ensemble combination. Each training problem has a smaller size compared to the size of the original problem. Consequently, better results are obtained. Another advantage of having fewer sensor nodes per subnetwork is the reduction in the transmission power and communication energy as the nodes become close to each other. The idea of treating localization problem as regression problem in-stead of a classification one was introduced in (Bhatti 2018). The algorithm is divided into two phases. The training dataset for localization in WSN contains of the feature vectors related to each anchor and its true location coordinates which are already known. The feature vector of an anchor node is composed of the RSS values of the signals received from other nodes as measured by that anchor node. In the second phase, the sensor nodes' coordinates are estimated. The input of the learned model is the target WSN nodes' feature vectors. The produced output is the nodes' coordinates. Comparing the extended feature vector (readings from anchor and sensor nodes) versus the reduced feature vectors (readings from anchor vectors only) showed that the extended features provide a better prediction accuracy.

Another localization technique was explained in (Kumar 2018) using fuzzy logic ML technique. The proposed technique was a hybrid Sugeno-Mamdani fuzzy system using RSS for localization which outperformed the traditional fuzzy logic. The authors then proposed the idea of cooperative localization. In such scheme, whenever an unknown node gets localized, it acts as anchor or landmark for the next iteration and transmits beacons to other unknown nodes. Since it has already been localized now and can broadcast beacons which contain its position, identity to be received by other unknown nodes. A low-complexity centroid-based scheme is proposed in (Kumar 2018). The resulting precision error is high. Hence, the authors used a FL algorithm to improve the nodes' location estimation based on FL weights. For more optimization, the consequences of unbalanced node placements in non-uniform networks are alleviated using PSO. This approach proved its efficiency in networks with few nodes and limited sensing coverage. As the number of nodes and/or the sensing coverage increase, integrating extreme machine learning techniques (as NN) with the centroid scheme showed better results in (Phoemphon et al. 2018).

6.4.2 Clustering

IoT systems are energy-constrained networks. Transmitting all the data packets to the sink node is inefficient and dramatically consumes the nodes' energy. A local aggregator, or a cluster head (CH), is used to improve the energy-efficiency by collecting the data from the cluster members within its vicinity and transmitting only the aggregation of the data to the sink node. Machine learning algorithms can help in deciding the number of clusters needed and electing the cluster heads.

An integrated approach in which clustering is performed using a SOM phase followed by a *k*-means phase was introduced in (Ahmadnezhad and Rezaee 2015). The SOM input parameters are the energy levels and the nodes' coordinates. The weight vectors of the SOM map units are selected nodes with maximum energy levels. Such maximum-energy nodes attract the nearest lower-energy nodes, thereby, creating energy-balanced clusters. Fuzzy logic techniques can also be used in electing cluster heads as proved in (Nayak and Devulapalli 2016). The authors of (Nayak and Devulapalli 2016) proposed a fuzzy logic clustering approach using the battery power, node mobility and node centrality as input parameters to a FL system that finds the probability of a node to serve as a cluster head. Simulation results showed that the fuzzy logic cluster head election system outperforms the well-known LEACH clustering protocol in terms of the network lifetime defined as the time until first node dies, last node dies or half the nodes die.

A cluster formulation method in which a node individually decides its ability to serve as a CH rather than executing an election process is presented in (Forster and Murphy 2006). This clustering method exploits Q-learning alongside a set of dynamic parameters such as the nodes' energy levels. A reinforcement learning technique was also used in (Soni and Shrivastava 2018) to implement an on-demand

mobile sink traversal. Recently, evolutionary computation algorithms are used showing enhancement in solving the clustering problem as in (Wang et al. 2018) where ACO-based approach is used. Likewise, Energy Centers Searching using Particle Swarm Optimization (EC-PSO) is proposed in (Wang et al. 2019). A geometric method is initially used to elect the CHs. Then, EC-PSO performs clustering when the nodes' energies start to be heterogeneous. EC-PSO elects the nodes close to the energy center to be CHs using an improved PSO technique that searches the energy centers. PSO algorithm increases the network lifetime as proved in (Yadav et al. 2018).

6.4.3 Routing

Designing a routing protocol for IoT systems is very challenging due to their nature of restricted processing, compact memory, and low bandwidth. Routing protocols address several issues including scalability, energy utilization, data coverage, and fault tolerance while optimizing their tradeoffs. Machine learning can effectively address this challenge as it continuously discovers the optimal routing paths that result in the best tradeoffs is the dynamically changing IoT networks. ML also reduces the complexity of a typical routing problem by breaking it down to subrouting problems that only consider the local neighbors of the nodes. Finally, ML effectively achieves the routing QoS requirements despite the use of computationally inexpensive algorithms and classifiers (Alsheikh et al. 2014).

A wireless routing protocol that uses SOM alongside a modified radial-based neural network was presented in (Hoomod and Jebur 2018). It starts with clustering the networks using SOM as previously explained. Then, an ANN will be responsible for finding the optimal path. However, the used ANN is modified by having the weights to the output layer computed and attuned using the parallel Moore-Penrose generalized pseudo-inverse which accelerates the learning process and accuracy. Taking time as a comparison metric, the proposed protocol outperformed the traditional Dijkstra in fixed and mobile topologies.

FL was also used in solving the routing challenge. In (Mottaghinia and Ghaffari 2018), two fuzzy logic-based systems were proposed to route data messages and specify their priority. When a source node encounters other nodes, it checks the data delivery probability of each node alongside the node's energy. A node is not considered as a potential router if either the residual energy and delivery probability is low, or both is low. Ultimately, the fuzzy system will select the best candidate for data transmission from the nodes' neighbors. The proposed approach shows its efficiency by increasing the data delivery rate and decreasing the associated delay.

Another approach combines FL (for clustering) and ACO (for routing) (Arjunan and Sujatha 2018). The node's residual energy, degree, centrality and distance to both the Base Station (BS) and neighboring nodes are used as inputs to the FL-based clustering algorithm which maximizes the network lifetime while balancing the nodes' energies. ACO is used to get the shortest paths. Also, periodic random choice

of paths occurs to exploit unused paths and to balance the nodes' energies. Again, FL is used in (Thangaramya et al. 2019) but this time for enhancing the NN to be used in discovering energy-efficient routes. The NN discovers new routes by investigating the consumed energy in the nodes and the routing patterns. The routes' weights are attuned by applying FL rules to reach the most efficient route.

6.4.4 Data Aggregation

Data aggregation is very crucial to reduce the IoT system power consumption. It combines and summarizes the data packets of several nodes properly at the cluster head. Data aggregation decreases the number of transmitted data packets, thereby, increases the bandwidth utilization and minimizes the energy consumption. In what follows, we demonstrate the power of machine learning techniques in this field.

First, (Pinto et al. 2014) proposed using genetic algorithms in information fusion to perform a trade-off between the Quality of Fusion (QoF) and efficiency by dynamically adjusting the sending probability. According to the effect of a node's input on the system performance, each node is given a reward that will decide whether to send data or fuse it with another node instead.

A priority-based data aggregation approach was introduced in (Ghate and Vijayakumar 2018). It works in supervised mode if the input data has class labels. For example, the health issue or the disease may be known previously in certain cases, hence, it can be used as a class label in the output vector. Alternatively, this approach may work in unsupervised mode with input data lacking class labels. A novel approach combining PCA and Angle Optimized Global Embedding (AOGE) was introduced to tackle the concept drift problem (Liu et al. 2017). In ML context, concept drift implies that the target variable, to be predicted by the model, has time-varying statistical properties that vary in unforeseen ways. Consequently, the prediction process results in less accurate predictions with time. AOGE takes advantage of several techniques. The projection variance of sampled data is first analyzed. Then, PCA is used to define the dispersion in the data. The principal components are then selected considering the maximized projection variance. Unlike PCA, AOGE analyzes the projection angle of sampled data to choose the principal components. Consequently, AOGE outperforms PCA when tested using real-life datasets with significantly noisy data. This implies that even though PCA and AOGE separately detect concept drift in a data stream, their combination is more effective and robust in detecting concept drift.

6.5 Machine Learning for IoT Performance Aspects

While the basic operational challenges are directly associated with functional behavior of IoT systems, performance aspects are mostly associated with performance enhancement. The performance enhancing requirements include fault detection, mitigation and controlling congestion provide quality of service and maintain security. This section sheds the light on the exploitation of machine learning in such performance-related aspects (summarized in Fig. 6.8).

6.5.1 Congestion Control

Congestion negatively impacts the performance of IoT applications as it causes packet losses, increases the encountered delays, wastes the nodes' energies and significantly degrades the IoT application fidelity. The purpose of IoT and WSN congestion control is to improve the network throughput and reduce the time of data

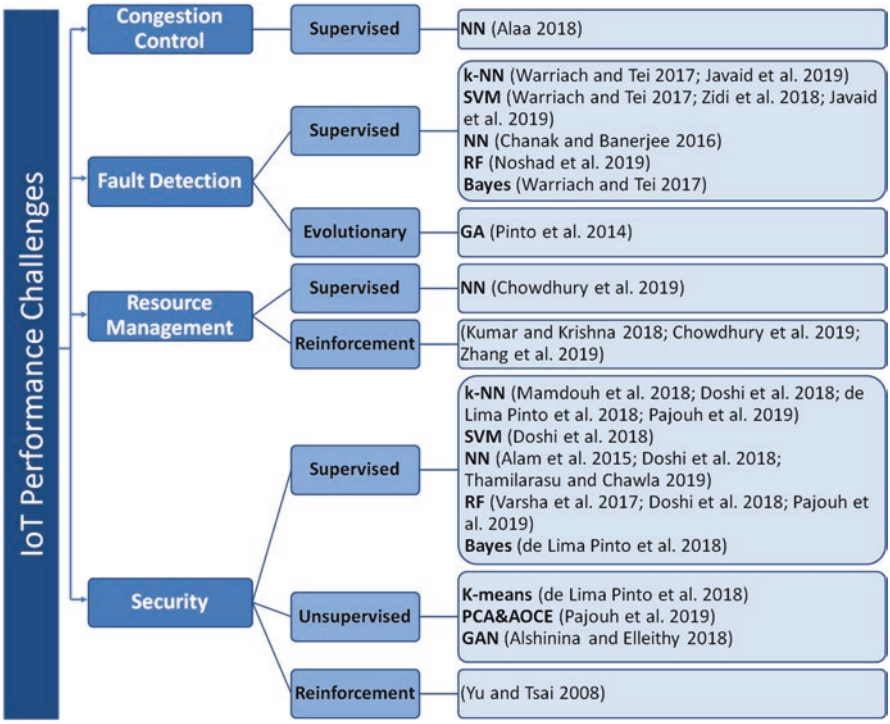

Fig. 6.8 Machine learning exploitation for enhancing IoT performance

transmission delay. Thus motivated, the authors of (Alaa 2018) proposed to have a congestion detection phase followed by a congestion monitoring phase. The system detects the congestion through measuring the data loss rate. The congestion monitoring system is simply an ANN for learning about the congestion scenarios to reduce and stop them before they even occur. This approach proved significant improvement when compared against the traditional case of not having congestion control.

6.5.2 Fault Detection

As explained in (Warriach and Tei 2017), faults occur when one or more of the IoT system characteristics or parameters deviate from their normal operation or value. Faults occur when node is faulty because of a physical damage, a low battery, communication interference, or environmental interference. An error is defined as an incorrect sensing of a state or event in the given space due to a fault. Faults can be classified into:

- Offset fault: This fault occurs when the data always differ from its expected value by a constant amount because of faulty calibration of the sensing module.
- Gain fault: This fault occurs when the rate of change of the sensed data in a period of time differs from its expected value.
- Stuck-at fault: This type of faults happens when the sensed data is constant and does not vary with time (zero-variance).
- Out-of-bounds fault: This fault happens when the values of the sensed data exceeds the normal operation bounds.

In (Warriach and Tei 2017), the fault detection problem is changed into a simple classification problem where the received data either belongs to a normal or a defected class. Three machine learning approaches were used for this purpose: *k*-NN, SVM and Naïve Bayes. *k*-NN has the least classification error in the least computing time followed by SVM. However, Naïve Bayes showed the worst performance. In (Zidi et al. 2018), the authors brought attention to a different kind of faults and how to solve it. This fault was the random fault that is defined as an instant error in which data is disturbed just for an instant of time. This error causes many positive or negative sharp peaks that influence the data of the sensors. These perturbations are very fast which makes them more difficult to detect. The authors proposed an SVM classifier for detecting instant errors which showed a high accuracy that reached 99%. Evolution of the traditional classifiers is proposed in (Javaid et al. 2019) by implementing Enhanced SVM (ESVM) which combines SVM and GA. Also, the authors implemented Enhanced KNN (EKNN) and Enhanced Recurrent Extreme Learning Machine (ERELM) which gives the most accurate results. Another classifier was introduced in (Noshad et al. 2019) which uses RF algorithm and shows better results compared to SVM and NN.

6.5.3 Resource Management

To satisfy the enormous resource demands of the various IoT applications, robust resource management techniques are needed to minimize the energy consumption and the response time. Since IoT systems are dynamic in nature, RL is one of the most suitable ML technique in IoT resource management as proposed in (Kumar and Krishna 2018). However, RL complexity increases with the increase in action pairs. Researchers combined NN and RL to introduce Drift Adaptive Deep Reinforcement Learning (DA-DRL) in (Chowdhury et al. 2019) to enhance traditional RL methods. Another scheduling technique was suggested in (Zhang et al. 2019) as Q-Learning Scheduling on Time Division Multiple Access (QS-TDMA) to improve the real-time reliability.

6.5.4 Security

As IoT systems are resource limited, securing such systems against security at-tacks presents an immense challenge. Several approaches for secure authentication in IoT systems through cloud computing exist (Kumari et al. 2018; Alam et al. 2015). However, the majority of contemporary IoT commercial devices suffer severe security flaws and vulnerabilities as shown in (Williams et al. 2017). That is why the demand for using ML techniques is rapidly growing to save such networks from different security attacks. Here, we discuss the major five IoT attacks (Mamdouh et al. 2018).

- Distributed Denial of Service (DDOS) Attack: In this cyber-attack, the attackers overload the system making it difficult to be used by its intended users by sending multiple requests to exceed its capacity, and therefore, crushing.
- Spoofing Attack: Is a cyber-attack where attackers aim to masquerade and deceive the system by pretending to be an authorized node to trick them in performing legitimate actions or giving up sensitive data.
- Malware Attack: Is a cyber-attack in which a malware or a malicious software performs activities on the victim's operating system, usually without the node's knowledge.
- User to Root (U2R): In this attack, the attacker attempts to escalate a user's privilege from being limited to become a super user or be able to access the root. This is achieved using stolen credentials or through a malware infection.
- Remote to Local (R2L): In this attack, the attacker imitates a legitimate user to gain remote access to a victim device.

In (Doshi et al. 2018), the authors detect DDOS attacks through capturing the data traffic and analyzing its features. It was proved that normal IoT traffic differs from DDOS traffic in terms of packet size, inter packet arrival, used protocol, the bandwidth and node/IP destination. Based on that observation, normal ML

classification techniques such as SVM, *k*-NN, ANN and Random Forest were used. It was shown that Random Forest and *k*-NN perform best. However, an updated technique was presented in (Thamilarasu and Chawla 2019) which combines ML and deep learning algorithms to form a deep neural network to detect DDOS attacks more precisely. NN is used with cooperation on cloud trace back technique to detect DDOS attacks as in (Alam et al. 2015).

Detecting spoofing attacks needs four main stages as illustrated in Fig. 6.9. ML techniques are commonly used in the feature detection and attack detection stages. In (de Lima Pinto et al. 2018), feature detection was achieved using a *k*-means algorithm and a *k*-NN classifier was proposed. Another technique was proposed in (Pajouh et al. 2019) that uses PCA in dimension reduction in addition to two classifiers: Naïve Bayes classifier followed by a *k*-NN classifier. This two-tier classification has high detection rates and accurate detection of the U2R and R2L hard-to-detect security attacks.

Malware detection was approached as a classification problem using random forest and *k*-NN in (Pajouh et al. 2019). An interesting approach for intrusion detection was shown in (Yu and Tsai 2008) in which each sensor node is equipped with an intrusion detection agent (IDA). As nodes cannot trust each other, IDAs do not cooperate. A Local Intrusion Detection Component (LIDC) is responsible for extracting the local features such as the packet delivery and collision rates, delays, number of neighbors, cost of routing and consumed energy. Meanwhile, Packet based Intrusion Detection Component (PIDC) infers if a suspected node is launching an attack on the host by analyzing the suspected node's packets and investigate the packets' RSS, the arrival rate of the sensed data and retransmission rate of the attacker's packets. Then, SLIPPER machine learning algorithm is used for detection.

Finally, securing WSN and IoT middleware using Generative Adversarial Networks (GANs) was proposed in (Alshinina and Elleithy 2018). The proposed approach is composed of two networks. A generator network that generates fake data that mimics the real sensed data and confuses the attacker by combining both the fake and real data. A discriminator network is then used to separate the fake data from the real data. This does not only protect data from adversaries but also improves the data accuracy compared to conventional techniques.

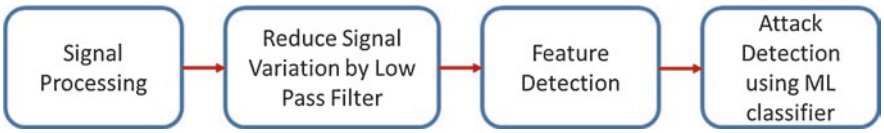

Fig. 6.9 The detection stages of spoofing attacks

6.6 Concluding Remarks

The unique nature of WSNs and IoT systems gives us no choice but to address their challenges and limitations through suitable tools and specified techniques. Here comes the need for machine learning techniques either supervised learning, unsupervised learning, reinforcement learning, evolutionary computation or fuzzy logic. All such techniques offer different solutions to most of the challenges. In this chapter, we have discussed these solutions for addressing the IoT basic operation challenges such as node localization, cluster formulation, routing and data aggregation. Moreover, we have discussed the role of machine learning in solving the performance-related challenges such as congestion control, fault detection, resource management and security. We conclude with the following remarks:

- Performance related aspects mainly exploit supervised learning techniques. Such challenges are handled as classification tasks where the algorithm needs to predict discrete value or identify the input data into a particular class. This requires prior knowledge and that is why supervised ML techniques are suitable here.
- Evolutionary techniques are used for solving basic operation rather than performance related aspects. Their objective is injecting new actions and measuring their effect e.g. by imitating ants in reaching their destination. This makes evolutionary technique not suitable for performance challenges which are typically modelled as classification tasks.
- Since fuzzy systems are capable of handling uncertainties and giving wide range of truth, they are recently being adopted for IoT routing and node localization.
- Resource management is solved using reinforcement learning (Q-learning technique). IoT systems are very dynamic. Managing their resources also needs a dynamic technique that always interacting with the surrounding environment to make the right immediate actions. Hence, RL comes as a perfect match here.

References

Ahmadnezhad, F., & Rezaee, A. (2015). Increasing the lifetime of wireless sensor networks by self-organizing map algorithm. *International Journal of Computer Networks and Communications Security, 3*(4), 156–163.

Alaa, M. (2018). Radial basis neural network controller to solve congestion in wireless sensor networks. *Iraqi Journal for Computers and Informatics, 44*(1), 53–62.

Alam, M., & Shakil, K. A. (2016). Big data analytics in cloud environment using Hadoop. In *International conferences on mathematics, physics & allied sciences*.

Alam, B., Doja, M., Alam, M., & Malhotra, S. (2013). 5-layered architecture of cloud database management system. *AASRI Procedia Journal, 5*, 194–199.

Alam, M., Shakil, K.A., Javed, M. S., & Ansari, M. (2015). Ambreen: Detect and filter traffic attack through cloud trace back and neural network. In *International conference of parallel and distributed computing*.

Ali, S. A., & Alam, M. (2016). A relative study of task scheduling algorithms in cloud computing environment. In *2nd international conference on contemporary computing and informatics (IC3I)*.

Ali, S. A., Affan, M., & Alam, M. (2019). A study of efficient energy management techniques for cloud computing environment. In *9th international conference on cloud computing, data science & engineering (Confluence)*.

Alsheikh, M., Lin, S., Niyato, D., & Tan, H. (2014). Machine learning in wireless sensor networks: Algorithms, strategies, and applications. *IEEE Communication Surveys and Tutorials, 16*(4), 1996–2018.

Alshinina, R., & Elleithy, K. (2018). A highly accurate deep learning based approach for developing wireless sensor network middleware. *IEEE Access, 6*, 29885–29898.

Arjunan, S., & Sujatha, P. (2018). Lifetime maximization of wireless sensor network using fuzzy based unequal clustering and ACO based routing hybrid protocol. *Applied Intelligence, 48*(8), 2229–2246.

Ayodele, T. (2010, February). Introduction to machine learning. In Y. Zhang (Ed.), *New advances in machine learning*. IntechOpen.

Banihashemian, S., Adibnia, F., & Sarram, M. (2018). A new range-free and storage-efficient localization algorithm using neural networks in wireless sensor networks. *Wireless Personal Communications, 98*(1), 1547–1568.

Bhatti, G. (2018). Machine learning based localization in large-scale wireless sensor networks. *Sensors, 18*(12), E4179.

Chanak, P., & Banerjee, I. (2016). Fuzzy rule-based faulty node classification and management scheme for large scale wireless sensor networks. *Expert Systems with Applications, 45*(C), 307–321.

Chowdhury, A., Raut, S., & Narman, H. (2019). DA-DRLS: Drift adaptive deep reinforcement learning based scheduling for IoT resource management. *Journal of Network and Computer Applications, 138*, 51–65.

de Lima Pinto, E., Lachowski, R., Pellenz, M., Penna, M., & Souza, R. (2018). A machine learning approach for detecting spoofing attacks in wireless sensor networks. In *IEEE international conference on Advanced Information Networking and Applications (AINA)*.

Doshi, R., Apthorpe, N., & Feamster, N. (2018). *Machine learning DDoS detection for consumer Internet of Things devices*. arXiv preprint arXiv: 1804.04159.

El Assaf, A., Zaidi, S., Affes, S., & Kandil, N. (2016). Robust ANNs-based WSN localization in the presence of anisotropic signal attenuation. *IEEE Wireless Communications Letters, 5*(5), 504–507.

Forster, A., & Murphy, A. (2006). CLIQUE: Role-free clustering with Q-learning for wireless sensor networks. In *IEEE international conference on distributed computing systems*.

Ghasempour, A. (2019). Internet of Things in smart grid: Architecture, applications, services, key technologies, and challenges. *Inventions, 4*(1), 22.

Ghate, V., & Vijayakumar, V. (2018). Machine learning for data aggregation in WSN: A survey. *International Journal of Pure and Applied Mathematics, 118*(24), 1–12.

Guo, Y., Sun, B., Li, N., & Fang, D. (2018). Variational bayesian inference-based counting and localization for off-grid targets with faulty prior information in wireless sensor networks. *IEEE Transactions on Communications, 66*(3), 1273–1283.

Habib, A., Arafat, M., & Moh, S. (2018). Routing protocols based on reinforcement learning for wireless sensor networks: A comparative study. *Journal of Advanced Research in Dynamical and Control Systems*, (14), 427–435. http://www.jardcs.org/backissues/abstract.php?archiveid=6166

Hoomod, H., & Jebur, T. (2018). Applying self-organizing map and modified radial based neural network for clustering and routing optimal path in wireless network. *Journal of Physics: Conference Series, 1003*, 012040.

Jafarizadeh, V., Keshavarzi, A., & Derikvand, T. (2017). Efficient cluster head selection using naïve bayes classifier for wireless sensor networks. *Wireless Networks, 23*(3), 779–785.

Jain, B., Brar, G., & Malhotra, J. (2018). EKMT-k-means clustering algorithmic solution for low energy consumption for wireless sensor networks based on minimum mean distance from base station. In *Networking communication and data knowledge engineering*. Springer.

Javaid, A., Javaid, A., Wadud, Z., Saba, T., Sheta, O., Saleem, M., & Alzahrani, M. (2019). Machine learning algorithms and fault detection for improved belief function based decision fusion in wireless sensor networks. *Sensors, 19*(6), 1334.

Kang, J., Park, Y., Lee, J., Wang, S., & Eom, D. (2018). Novel leakage detection by ensemble CNNSVM and graph-based localization in water distribution systems. *IEEE Transactions on Industrial Electronics, 65*(5), 4279–4289.

Khan, S., Shakil, K. A., & Alam, M. (2016). Educational intelligence: Applying cloud-based big data analytics to the Indian education sector. In *2nd international conference on contemporary computing and informatics (IC3I)*.

Khan, S., Shakil, K. A., Ali, S. A., & Alam, M. (2018). On designing a generic framework for big data-as-a-service. In: *IEEE international conference on advanced research in engineering sciences*.

Khan, S., Shakil, K. A., & Alam, M. (2019a). PABED – A tool for big education data analysis. In *20th IEEE international conference on industrial technology*.

Khan, S., Liu, X., Ara Shakil, K., & Alam, M. (2019b). Big data technology – Enabled analytical solution for quality assessment of higher education systems. *International Journal of Advanced Computer Science and Applications (IJACSA), 10*(6). ESCI/Scopus.

Khan, S., Arshad Ali, S., Hasan, N., Ara Shakil, K., & Alam, M. (2019c). Big data scientific workflows in the cloud: Challenges and future prospects. *Cloud Computing for Geospatial Big Data Analytics*, 1–28.

Khan, S., Shakil, K. A., Alam M. (2019d). Big data computing using cloud-based technologies: Challenges and future perspectives. *Networks of the Future: Architectures, Technologies and Implementations*.

Kim, W., Park, J., Yoo, J., Kim, H., & Park, C. (2013). Target localization using ensemble support vector regression in wireless sensor networks. *IEEE Transactions on Cybernetics, 43*(4), 1189–1198.

Kotha, H., & Gupta, V. (2018). IoT application – A survey. *International Journal of Engineering & Technology, 7*, 891–896.

Kumar, A. (2018). *A hybrid fuzzy system based cooperative scalable and secured localization scheme for wireless sensor networks*.. International Journal of Wireless & Mobile Networks (Vol. 10, pp. 51–68).

Kumar, T., & Krishna, P. (2018). Power modelling of sensors for IoT using reinforcement learning. *International Journal of Advanced Intelligence Paradigms, 10*(1–2), 3.

Kumari, A., Abbasi, M. Y., Kumar, V., & Alam, M. (2018). The cryptanalysis of a secure authentication scheme based on elliptic curve cryptography for IOT and cloud servers. In *IEEE International Conference on Advances in Computing, Communication Control and Networking (ICACCCN)*.

Li, Y., & Parker, L. (2014). Nearest neighbor imputation using spatial–temporal correlations in wireless sensor networks. *Information Fusion, 15*, 64–79.

Liu, S., Feng, L., Wu, J., Hou, G., & Han, G. (2017). Concept drift detection for data stream learning based on angle optimized global embedding and principal component analysis in sensor networks. *Computers and Electrical Engineering, 58*, 327–336.

Malhotra, S., Doja, M. N., Alam, B., & Alam, M. (2018). Generalized query processing mechanism in cloud database management system. In *Big data analytics* (pp. 641–648). Singapore: Springer.

Mamdouh, M., Elrukhsi, M., & Khattab, A. (2018). Securing the Internet of Things and wireless sensor networks via machine learning: A survey. In *IEEE International Conference on Computer and Applications (ICCA)*.

Mehmood, A., Lv, Z., Lloret, J., & Umar, M. (2017). ELDC: An artificial neural network based energy-efficient and robust routing scheme for pollution monitoring in WSNs. *IEEE*

Transactions on Emerging Topics in Computing, 1–1. https://ieeexplore.ieee.org/abstract/document/7859382/citations#citations

Miljković, D. (2017). Brief review of self-organizing maps. In *IEEE international convention on information and communication technology, electronics and microelectronics (MIPRO)*.

Mottaghinia, Z., & Ghaffari, A. (2018). Fuzzy logic based distance and energy-aware routing protocol in delay-tolerant mobile sensor networks. 100(3): 957–976.

Navani, D., Jain, S., & Nehra, M. (2017). The Internet of Things (IoT): A study of architectural elements. In *13th international conference on Signal-Image Technology & Internet-Based Systems (SITIS)*.

Nayak, P., & Devulapalli, A. (2016). A fuzzy logic-based clustering algorithm for WSN to extend the network lifetime. *IEEE Sensors Journal, 16*(1), 137–144.

Noshad, Z., Javaid, N., Saba, T., Wadud, Z., Saleem, M., Alzahrani, M., & Sheta, O. (2019). Fault detection in wireless sensor networks through the random forest classifier. *Sensors, 19*(7), 1568.

Pajouh, H., Javidan, R., Khayami, R., Dehghantanha, A., & Choo, R. (2019). A two-layer dimension reduction and two-tier classification model for anomaly-based intrusion detection in IoT backbone networks. *IEEE Transactions on Emerging Topics in Computing, 7*(2), 314–323.

Phoemphon, S., So-In, C., & Niyato, D. (2018). A hybrid model using fuzzy logic and an extreme learning machine with vector particle swarm optimization for wireless sensor network localization. *Applied Soft Computing, 65*, 101–120.

Pinto, A., Montez, C., Araújo, G., Vasques, F., & Portugal, P. (2014). An approach to implement data fusion techniques in wireless sensor networks using genetic machine learning algorithms. *Information Fusion, 17*, 90–101.

Sam, S. (2016). Internet of Things' connected devices to triple by 2021, reaching over 46 billion units. Juniper Research.

Sethi, P., & Sarangi, S. (2017). Internet of Things: Architectures, protocols, and applications. *Journal of Electrical and Computer Engineering, 1*, 1–25.

Shakil, K. A., Zareen, F. J., Alam, M., & Jabin, S. (2017). BAM health cloud: A biometric authentication and data management system for healthcare data in cloud. *Journal of King Saud University – Computer and Information Sciences* (in press). https://www.sciencedirect.com/science/article/pii/S1319157817301143

Soni, S., & Shrivastava, M. (2018). Novel learning algorithms for efficient mobile sink data collection using reinforcement learning in wireless sensor network. *Wireless Communications and Mobile Computing*, 2018:7560167, 13 pages.

Sun, B., Guo, Y., Li, N., & Fang, D. (2017). Multiple target counting and localization using variational Bayesian EM algorithm in wireless sensor networks. *IEEE Transactions on Communications, 65*(7), 2985–2998.

Sun, Y., Zhang, X., & Wang, X. (2018). Device-free wireless localization using artificial neural networks in wireless sensor networks. *Wireless Communications and Mobile Computing*, *2018*, 4201367, 8 pages.

Thamilarasu, G., & Chawla, S. (2019). Towards deep-learning-driven intrusion detection for the internet of things. *Sensors, 19*(9), 1977.

Thangaramya, K., Kulothungan, K., Logambigai, R., Selvi, M., Ganapathy, S., & Kannan, A. (2019). Energy aware cluster and neuro-fuzzy based routing algorithm for wireless sensor networks in IoT. *Computer Networks, 151*, 211–223.

Umarikar, A. (2003). *Fuzzy logic and brief overview of its applications*. University Västerås Suecia.

Van der Meulen, R. (2017). Gartner says 8.4 billion connected things will be in use in 2017, up 31 percent from 2016. Garther Research.

Varsha, S., Shubha, P., & Avanish, T. (2017). Intrusion detection using data mining with correlation. In *2nd international conference for Convergence in Technology (I2CT)*.

Vashi, S., Ram, J., Modi, J., Verma, S., & Prakash C. (2017). Internet of Things (IoT): A vision, architectural elements, and security issues. In *IEEE international conference on IoT in Social, Mobile, Analytics and Cloud (I-SMAC)*.

Wang, Z., Liu, H., Xu, S., Bu, X., & An, J. (2017). Bayesian device-free localization and tracking in a binary RF sensor network. *Sensors, 17*(5), 1–21.

Wang, J., Cao, J., Sherratt, R., & Park, J. (2018). An improved ant colony optimization-based approach with mobile sink for wireless sensor networks. *The Journal of Supercomputing, 74*, 6633–6645.

Wang, J., Gao, Y., Liu, W., Sangaiah, A., & Kim, H. (2019). An improved routing schema with special clustering using PSO algorithm for heterogeneous wireless sensor network. *Sensors, 19*(3), 671.

Warriach, E., & Tei, K. (2017). A comparative analysis of machine learning algorithms for faults detection in wireless sensor networks. *International Journal of Sensor Networks, 24*(1), 1–13.

Williams, R., McMahon, E., Samtani, S., Patton, M., & Chen, H. (2017). Identifying vulnerabilities of consumer Internet of Things (IoT) devices: A scalable approach. In *IEEE international conference on Intelligence and Security Informatics (ISI)*.

Yadav, A., Kumar, S., & Vijendra, S. (2018). Network life time analysis of WSNs using particle swarm optimization. *Elsevier, 132*, 805–815.

Yu, Z., & Tsai, J. (2008). A framework of machine learning based intrusion detection for wireless sensor networks. In *IEEE international conference on sensor networks, ubiquitous, and trustworthy computing*.

Zhang, B., Wu, W., Bi1, X., & Wang, Y. (2019). A task scheduling algorithm based on Q-learning for WSNs. *The Abel Prize*, 521–530.

Zidi, S., Moulahi, T., & Alaya, B. (2018). Fault detection in wireless sensor networks through SVM classifier. *IEEE Sensors Journal, 18*(1), 340–347.