IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 9 April 2023, accepted 25 April 2023, date of publication 3 May 2023, date of current version 6 June 2023.

*Digital Object Identifier 10.1109/ACCESS.2023.3272691*

RESEARCH ARTICLE

# Indexing in WoT to Locate Indoor Things

**MUHAMMAD REHAN FAHEEM[1], TAYYABA ANEES[1], MUZAMMIL HUSSAIN[1], ALLAH DITTA[2], HANI ALQUHAYZ[3], AND MUHAMMAD ADNAN KHAN[4,5,6]**

[1]School of Systems and Technology, University of Management and Technology, Lahore 54770, Pakistan
[2]Department of Information Sciences, Division of Science and Technology, University of Education, Lahore 54000, Pakistan
[3]Department of Computer Science and Information, College of Science, Zulfi, Majmaah University, Al-Majma'ah 11952, Saudi Arabia
[4]School of Information Technology, Skyline University College, University City Sharjah, Sharjah, United Arab Emirates
[5]Riphah School of Computing and Innovation, Faculty of Computing, Riphah International University, Lahore Campus, Lahore 54000, Pakistan
[6]Department of Software, Faculty of Artificial Intelligence and Software, Gachon University, Seongnam 13120, Republic of Korea

Corresponding author: Muhammad Adnan Khan (adnan@gachon.ac.kr)

This work was supported by the Deanship of Scientific Research at Majmaah University under Project R-2023-376.

**ABSTRACT** The Web of Things (WoT) is an enhanced form of the Internet of Things (IoT) that has changed the trend of life nowadays. Due to IoT, life is transformed into smart life, such as smart buildings, smart vehicles, smart agriculture, smart businesses, etc., by connecting a certain number of things to the internet. Many people are now working on ways to locate indoor things to interact and exchange data between smart things and web services and apps, which is called "WoT," or "Web of Things." To interact and exchange the data, researchers need a search engine on WoT. However, locating indoor things in the Web of Things (WoT) remains a challenge due to the lack of a unified indexing system. In this research, we propose a novel approach to index indoor things in the WoT by leveraging machine learning and web technologies. Our approach includes a data preprocessing step, where we extract relevant features from the sensor data, followed by a clustering algorithm to group similar devices. We then use a semantic model to assign meaning to the clusters and develop a search engine to enable efficient searching of indoor things. Our proposed approach improves the accuracy and efficiency of locating indoor things in the WoT, paving the way for new applications in smart homes, healthcare, and industrial automation.

**INDEX TERMS** Naïve Bayes, cluster, crawling, indexing, indoor things, things indexer, symbolic data.

## I. INTRODUCTION

The Internet of Things (IoT) has become the preeminent technology in the last few years. The internet of things (IoT) enables sensors to communicate with devices such as electric cars, smartphones, and wearable devices, among others, via the internet [1]. IoT plays an important role in every aspect of life, such as smart health, smart education, smart agriculture, etc. With time, the number of IoT devices has increased [2]. Due to this, users are facing issues while searching for indoor IoT devices.

Locating indoor things accurately is one of the trendiest problems in the field of IoT. Only outdoor location-based services can be provided by Global Navigation Satellite Systems, such as the Global Positioning System. Numerous IPS systems have been proposed so far to provide indoor location-based services for intricate confined areas like hospitals, airports, retail malls, and more. Indoor location-based services also aid in resource management, which entails efficient resource deployment, tracking, and location. There are two types of technology for IPS solutions: infrastructure-based and infrastructure-free. While infrastructure-free solutions typically rely on Ultra-Wideband, Bluetooth, RF, and Radio Frequency Identifier-based technologies, infrastructure-based solutions are typically based on motion sensors, Wi-Fi, magnetic fields, and visualization techniques. Infrastructure-free solutions are preferred because they are more affordable and convenient to use than infrastructure-based ones, which necessitate costly pre-installation and setup of particular equipment in a certain location [3].

In general, Wi-Fi-based approaches can be divided into 1) techniques that need calibration and 2) techniques that do not need calibration. Due to the presence of LOS and

The associate editor coordinating the review of this manuscript and approving it for publication was Yiming Tang.

NON-line-of-sight situations, an interior environment is referred to as a hybrid environment [4]. The intricacy of an interior environment limits the use of trilateration or triangulation-type calibration-free techniques. When adopting IPS methods that use a time difference of arrival and the time of arrival, these results in a transmission error on several paths. They are additionally made more challenging by synchronizing the precise short-range flight time measurements with receiver and transmitter-side times [5]. On the other hand, the angle of arrival technique needs sophisticated gear to calculate angles. To solve the issue of time synchronization, the received signal strength indicator value is employed for position estimation. The received signal strength indicator of APs received at a specific area is used in the calibration-based Wi-Fi fingerprinting technique. One of the most promising methods for locating in a Wi-Fi context is fingerprinting. Furthermore, it is unaffected by the specifications for indoor maps, multipath effects, and the positioning of transmitters. The offline training step for fingerprinting includes an environmental survey before deployment, and specific reference sites provide a calibrated RSSI database. The estimation of the position, known as the online phase of fingerprint matching, then uses this calibration database [6]. This method's general usage is constrained by the hard effort of calibrating the fingerprinting database through the survey. Numerous methods have been proposed so far to reduce the amount of effort needed during the database's calibration phase. Such strategies mostly comprise crowd-sourcing techniques, sparse reference point interpolation, and other techniques [7], [8], [9], [10], [11], [12]. As an alternative to the time-consuming site survey required for the calibration of the Wi-Fi fingerprint database, a calibration-free path loss model technique is also suggested in which position information of Aps and maps is utilized. Similar to this, other fingerprinting technique variations have been put up to date.

For this purpose, to locate indoor things there searchers have been working on the Web of Things (WoT) for the last few years [13]. The Web of Things (WoT) is the growing paradigm of the Internet of Things (IoT) that allows communication between physical objects (Things) through web services and applications [14]. Physical objects such as airplanes, cars, keys, bulbs, etc. are all Things as shown in Figure 1.

WoT search engine faced different challenges and issues, such as the description of Things in WoT is complex and their data is dependent on spatial and thematic dimensions. The status of data changes randomly and unexpectedly due to the loss of connection, physical problems in the sensor node, and changes in the location of Thing [15]. The data of Things are heterogonous and dynamic, as Things are deployed in distributed geographical areas. There are different ways to represent the discovered data, such as CSV, JSON, XML, JSON-LD, etc. [16].

This paper aims to explain how to optimize the efficiency of Data discovery and indexing using machine learning tools/algorithms while locating indoor things. As the data of

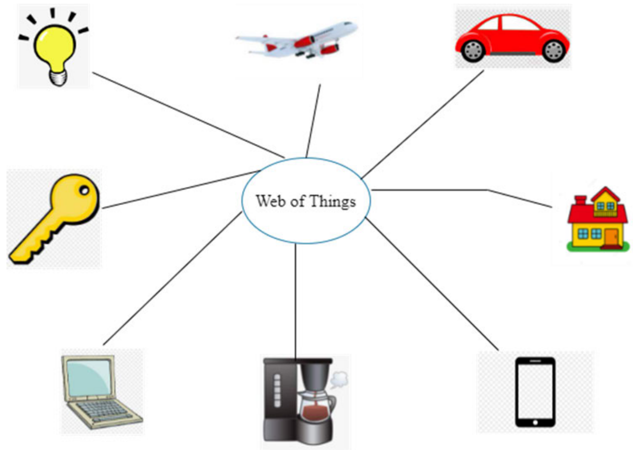



**FIGURE 1.** Things that are searched using the search engines based on WoT.

WoT is numeric as well as streaming and dynamic, also the data is dependent on time and location. So to cluster the data set based on the nature of the data, we have to use the machine learning algorithm to maximize the efficiency of clustering the data. And then used the NaïveBayes classifier to classify the cluster data for indexing. The following here described the contributions and advantages of the proposed model over the existing methods.

Due to the significant amount of data and variety of Things, indexing indoor Things in WoT search engine is a difficult process. To reflect the developments in this field, there are a few significant contributions that we made in this paper:

1. Semantic indexing: Semantic indexing is one of the most significant contributions made to the indexing of indoor things in WoT search engine. In this process, we entail detecting and categorizing the things based on their functionalities and connections to other things. This can help to improve the accuracy and relevance of search results and enable users to find the devices and data they need.

2. Cluster the dataset based on the nature of the data

3. Classification of Things to identify as Indoor things or outdoor things

4. Optimize the efficiency of the proposed WoT search engine.

Advantages of the Proposed Model:

The advantages of the proposed model with the existing methods are as follows:

1. The proposed model formats the data into the latest formatting techniques which are easily recognized by the browser. While the existing methods use the WSDL, micro-formats, and DPWS metadata are obsolete and search engines do not understand these formats.

2. The proposed models perform the 6 parameters of Keyword based indexing while the existing method performs one or two parameters of keyword-based indexing.

3. The authors have classified location-based indexing into two categories to address issues related to heterogeneity and scalability in spatial indexing. These categories are relative

location, which includes detailed information such as street number, country, and postal code, and coordinates, which consist of Latitude and Longitude. Both relative location and coordinates are utilized to accommodate the dynamic nature of devices, unlike existing methods that do not incorporate such steps.

4. The proposed model performs hybrid indexing, keyword-based indexing, and spatial indexing while the existing methods may use one indexing technique at a time.

### A. BACKGROUND KNOWLEDGE

This section describes the data discovery, indexing, clustering, and Indoor things in detail to give the background knowledge about the WoT search engine basics elements.

#### 1) DATA DISCOVERY

Data discovery is a necessary phase in the process of discovering Things. For this purpose, initially, the data or resources are characterized as a description and then discovered utilizing specific protocols such as HTTP. Two distinct methods are currently available for data discovery. In one of the methods, the data are connected directly through the use of some programs where the resources are discovered within the restricted range. This makes it extremely difficult when searching for actual Things because physical Things require data/resources to be publically accessible, or some mechanism is at least required to make resources more accessible at a lower cost. In the second method, Machine to Machine communication system is utilized to discover the data or resources, in which the particular machine seeks data or requests information from another machine.

Several sources are available to discover the Things having certain characteristic features. Data discovered for the Internet of Things is both centralized and decentralized. The term ‘‘centralized discovery’’ refers to the collection of data via APIs or other applications. Decentralized data collection refers to the availability of the data via visits to the separate sites of Things. The data is heterogeneous because IoT/WoT data originates from a variety of sources and is stored in a variety of formats thus the structure of data varies as well [14]. Data in the form of Things are characterized as structured, unstructured, semi-structured, and high-dimensional data. Structured data is organized in the form of tables. Unstructured data is represented by symbols. The term ‘‘semi-structured’’ refers to the data that is in the form of a description. The term ‘‘high-dimension’’ refers to data that is large in volume.

In this study, data was reached and accessed through a variety of sources, such as environmental sensor data from Sensor Web APIs and energy sensor data, home sensor data from W3C APIs.

According to the author [17], IoT data might be symbolic, continuous, quantitative, discrete, static, or streaming in nature. According to the author [18], the data generated by the Internet of Things is numerical, discrete, and digital. Table 1 below summarizes the data type of IoT data and table 2 summarizes the characteristics of IoT data [14].

**TABLE 1.** Nature of IoT data.

| Data Type | Description |
|---|---|
| Numeric | If the data of IoT/WoT is in the form of numerical values then its datatype is numeric. e.g temperature of the room is "20 C". |
| Symbolic | If the data of IoT/WoT is in the textual or string form then its datatype is symbolic e.g temperature of the room is "hot". |
| Discrete | The data of IoT/WoT consist of limited values and its data type is discrete. Discrete value is numeric or symbolic as well. |
| Streaming | If the data of IoT/WoT changes in real-time then its type is streaming e.g. the data mobile sensors are streaming as mobile sensors continuously change their location. |
| Continuous | If the data of IoT/WoT is collected from measurements i.e. by observing then its datatype is continuous e.g. the sensors on roads that continuously observes the traffic of vehicles on road. |
| Digitized | If the data of IoT/WoT is in the digital form then its type is digitized e.g. the coordinates of Thing are in the digitized form. |
| Static | If the data of IoT/WoT does not change, then its data type is static e.g. the gas sensors which do not change their location the example of static sensors. |

**TABLE 2.** Characteristics of IoT data.

| Characteristics | Description |
|---|---|
| Volume | As the number of sensors and actuators increased it may result in the production of huge amounts of data. The volume of IoT data grows rapidly due to the growth in the IoT industry. |
| Variety | The data of IoT is of different varieties such as numeric, audio, video, text, etc. As well as the format of generated data is also different such as data formatted in tables, records, XML, etc. |
| Distribution | The IoT data is distributed over different geographical locations because the sensors or actuators are distributed and arranged in different geographical areas. |
| Veracity | It represents the quality & availability of IoT/WoT data. The quality of data depends upon the availability of resources/devices from where the data is collected. The availability of IoT data depends on the battery life, mobility, etc. due to which data changes randomly. As the data changes then the accuracy of the data also changes which affects the quality of the data. |
| Dynamicity | It means that the data of IoT/WoT continuously changes due to the change in the position of Thing e.g. fight tracking system. |
| Value | It means that the data of IoT/WoT is useful for the user or not. As the user are demanding useful information instead of raw data. If a user gets useful information then the value of data is also high. |
| Spatio-Temporality | It means that the data of IoT/WoT is dependent on the current location of Things temporarily as the Thing changes its current location to another location with time. |

#### 2) INDEXING

The most central component of any search engine is indexing without which no search engine can perform. Three ways of indexing are in the Web of Things (WoT), which are keyword-based indexing, spatial-based indexing, and hybrid indexing. Some research has been carried out on these indexing approaches and many researchers are still in the process.

References [20], [21], [22], and [23] have worked on keywords based indexing. References [20] and [21] was comprised of Name and description-based indexing. Similarly, [22] and [23] presented a search engine with only description-based indexing. Meanwhile, [24] proposed an indexing approach that performs indexing on behalf of Services.

In the same way, many researchers have been carried out on spatial-based indexing such as [15], [25], and [26]. An indexing approach proposed by [26] worked on Latitude and longitude. If the sensor changes its position then the proposed mechanism could not find the sensor.

Similarly, an indexing approach by [15] was not able to locate the change in the sensor. Although its mechanism was to temporarily save the location of the sensor. The indexing approach given by [25] was IR- tree-based indexing. This approach retrieves the documents with the help of keywords, a spatial approach but in this process, it took more time than usual.

Hybrid Indexing is also being research topic for many researchers [28, 29, 30, 31, and 32]. The hybrid approach by [29] worked for fixed-location devices only. Thus useless for devices that change their location. In comparison, another hybrid approach by [27] was also presented but it lacks the performance of finding an increased number of the sensor.

References [28] and [30] also developed a hybrid approach working on SPARQL and Node.js but they're also having a drawback in that these approaches could not locate the dynamic nature of devices.

3. Clustering

Clustering is a process in which a system combines identical objects. We have a lot of clustering approaches that we can apply to such data having large volumes as well as with different natures. For example, the data of a sensor has a large volume and the nature of the data is also different as some data is numeric, some are symbolic data, and some is continuous data. The clustering approach which we can apply in IoT or WoT includes hierarchical clustering, density-based clustering, Grid base clustering, centroid-based clustering, similarity-based clustering, co-clustering, and partition clustering [32].

- Hierarchical Clustering:

It is that clustering that clusters the data into tree form. For example, if we want to present the taxonomy of some object, in that case, we use hierarchical clustering for the dataset.

- Density Based Clustering

This type of clustering is applied to the dataset we need to cluster by shape or size.

- Grid Based Clustering

It is that clustering in which we cluster the data based on space, such as how much the data space will take up in memory.

- Centroid Clustering

This type refers to the clustering of those data sets which are close to each other such as data sets with the same location, same size, or having almost the same memory.

- Partition Clustering

That clustering which divides the dataset based on similar features is partition clustering. It starts by finding the similarity and dissimilarities between datasets and then grouping similar data in one cluster and dissimilar data into another cluster.

4. Indoor Things

The Internet of Things (IoT) is a rapidly expanding technology whose core concept is to connect all physical objects. Devices of IoT can be explained by the fact that data mining has repeatedly demonstrated how to make IoT systems smarter. An author [33], presents the review of published research on IoT-based applications in the Environment, health, and safety industries with an emphasis on building, health care, infrastructure management, and energy. Another author [34] used machine learning techniques to incorporate face recognition, which could help the blind use smartphones to explore the environment more easily. The author in [35], provided a method for merging noisy fingerprints with distance constraints for indoor localization.

Orienting people in complicated settings and moving them to other locations is a difficult task today in megacities. Due to the advanced nature of electronic technology instruments, Japanese authorities have pushed to deploy mobile phones and navigation systems for people with visual impairments. References [36], [37] provides a thorough study of various indoor locating methods along with their operational technologies. Agents there use spatial and temporal synergy to gather data from diverse sources and determine their placements.

Reference [38] presents a well-known paradigm for the theoretical research and interpretation of information fusion in networks. We thus concentrate on the most recent advancements that are highly pertinent to the current research. It is important to note that, depending on the particular application circumstances; navigation and localization might be coupled or uncoupled. The latter technique may not always be used in the former.

The Cell ID is the simplest localization technique available in cellular networks (which is Cell-of-origin). Other techniques employ the angles of radio waves or time (distance). Some of them, such as AoA (Angle of Arrival), TDoA (Time Difference of Arrival), and ToA (Time of Arrival), are better known than others. Without calculating range, IMES may effectively create 3D location data while employing a transmitter unit. Additionally, IMES is quite accurate, especially in enclosed spaces. In contrast, GPS, A-GPS, and cell towers are unable to provide data of comparable quality in these types of settings. One of the short-range radio technologies used for indoor positioning is ultra-wideband. As an alternative to Wi-Fi and Bluetooth technology (Receive Signal Strength

Indicator), positioning can be accomplished using the transit time approach (also known as Time of Flight). This method will calculate the light running time between a thing and many receivers. Similarly, cell-id and triangulation, which are employed by Wi-Fi and Bluetooth technologies, respectively, are not new. The volume of access points and the location-positioning algorithms to be used are the two factors that determine the precision that is ultimately attained.

A real issue in a pervasive smart environment is the monitoring of worker movements, shipment tracking, and other building material-based tracking. The wireless sensor network (WSN) has contributed to the tracking scenario's uniqueness. The difficulties encountered in this work relate to an efficient 2-dimensional mobile system tracking and determining a potential forecast of the object's precise location in the sensing area. The creation of an indoor-based WSN with wireless sensor nodes employs a location-sensing technology based on RSSI. The sensing region is categorized as shells, and the Markov model is used to assess the node's mobility. The suggested algorithm has been put through testing with a range of IoT-friendly speeds. Real-world research demonstrates the viability of the suggested two-dimensional algorithms. The findings were produced to demonstrate the object's precise location and minimum location inaccuracy [49].

The use of Internet of Things (IoT) devices in settings like homes and offices to study human behavior has gained popularity. Such devices may be used in a lab or "in the wild" in uncontrolled settings. With the latter, behavioral data can be gathered without being tainted by the artificiality of a lab trial. In comparison to lab tests, using IoT devices in regular settings also has the advantages of lower costs and less disruption to participants' normal routines, which aids in attracting them to the study. However, having an IoT infrastructure that can be quickly and easily implemented and from which real-time data can be collected securely is crucial in this situation. In this paper, the author introduces an Internet of Things testbed that enables real-world experimentation for extensive social research on indoor activities using situation-aware applications and real-time monitoring. The testbed has a short setup time, deployment flexibility, the ability to integrate a variety of IoT devices, durability, and scalability [50].

To gather data on underwater emergencies, this article suggests an underwater emergency communication network (UECN) supported by several UCLs and autonomous underwater vehicles (AUV). This article suggests a UECN to efficiently achieve an energy-efficient underwater emergency response and harness the benefits of various underwater communication lines (UCL). The researchers choose the best emergency response mode (ERM) for each underwater sensor node employing greedy searching and reinforcement learning to identify the "isolated" USNs (IUSN) after which IUSNs may be identified. Multiobjective optimization achieves the best tradeoff between response efficiency and energy consumption [51].

Attempts have been made to enhance the performance of Underwater Wireless Sensor Networks (UWSNs), but issues with void communications, packet collisions, and energy efficiency still exist. Researchers provide a unique Swarm Intelligence-based routing approach for energy and QoS-efficient data transmission from the underwater sensor node to the surface sink. To address the issues at hand, the protocol Energy Optimization utilizing Routing Optimization (EORO) is suggested. To select the best forwarder node for UWSN data delivery, they develop Effective Fitness Function-based Particle Swarm Optimization (EFF-PSO). In EORO, the intended source node first uses location data to identify forwarding relay nodes. The best relay node is then chosen using the EFF-PSO algorithm that takes a wide range of factors into account. These factors include residual energy, packet transmission speed, node connectedness, and distance. By preventing packet collisions, these values are specifically set to lower energy usage, delay, and throughput. Throughput, energy use, delay, and Packet Delivery Ratio were all higher with EORO than with underlying routing systems (PDR) [52].

### B. MOTIVATION

The motivation of the proposed method is to develop more efficient and effective ways to manage and access the growing number of connected devices in indoor spaces. With the rapid increase of Internet of Things (IoT) devices, locating and accessing specific devices or data streams can become challenging, especially in complex indoor environments. This aims to develop better indexing mechanisms and tools for the WoT, researchers can enable more seamless integration of devices and sensors into existing networks and systems. This can help to improve the overall performance and usability of IoT solutions, as well as support the development of new applications and services that rely on real-time data from indoor environments.

Furthermore, advances in indexing technology could lead to significant improvements in a wide range of fields, from healthcare to manufacturing, by enabling more precise and accurate monitoring of indoor environments and processes. This could lead to improvements in efficiency, productivity, and safety, while also enabling new insights and discoveries in various domains.

## II. STATE-OF-THE-ART

A lot of work has been done to explore and explain IoT data clustering from which below is briefly described relevant research.

One researcher [39] applied partition-based clustering on those IoT devices that were connected to the network traffic. This research used the K-means algorithm to cluster the IoT devices into partitions of weather stations, motion sensors, nest smoke alarms, and TP-Link Smart Plug. Similarly, another research [40] was carried out for partition clustering utilizing the K-means algorithm and additionally indexing the clustered data using a Binary tree based on containers at the cloud clusters fog computing level (B3CF).

Another research [41] clusters the data of wireless networks in the health sector by applying the hierarchical cluster technique of Jarvis Patrick clustering that divides the sensor nodes using the Gaussian Process Regression function. Jarvis Patrick Clustering reviews the sensor nodes after their division and then groups them based on bandwidth and energy.

One research [42] applied hierarchical clustering on network traffic control devices in which a heuristic-based clustering algorithm was utilized that works in a bottom-to-top manner. At first, the researcher calculated the sensor node range of devices and made cluster nodes of those having high sensor node range and those having low sensor node range arranged in a hierarchy.

Research conducted by [43] clusters the data of network traffic devices by grid-based clustering in which those devices with the same memory space were merged into one group based on the spatial configuration of network traffic devices. The algorithm used in this study was force based clustering algorithm which initially calculates the distance between two devices and if the distance is less or equal between these then the algorithm checks the apps of these devices and places the same spaced devices in one group.

Another researcher used the data from sensor devices placed at the University of California at Irvine and applied density-based clustering. For this purpose, the study utilized K-Prototype (KP). KP works to find the frequency of occurrences of all attributes of each device, and then groups the devices depending on the attribute with the highest to lowest occurring frequency [44]

References [45] cluster the data using K-means. The authors selected the Description attribute from the dataset and calculated its length. Those datasets having the same length were clustered in one group. Similarly, [46] utilizes k-means clustering those devices in one group which has a close distance. To calculate the distance the study used the Euclidian Distance technique.

Due to the numerous uses of location-based services and localization-based computing, the author [55] uses location-based services and localization-based computing. In localization systems, information about the targets' positions is crucial. One of the most well-known and widely utilized technologies for missile guidance and outside localization systems is GPS. To design new systems that may employ various technologies and methodologies, localization systems are used to locate or track people or objects. For instance, satellite systems with worldwide coverage, such as GPS, an assisted-global positioning system (AGPS), global navigation satellite systems (GNSS), and assisted-global navigation satellite systems, have been used to estimate outside positioning, tracking, and navigation (AGNSS).

Author [56] described the challenges and opportunities of real-time data enabled by IoT devices. According to the author [56], a "smart" society is being created globally where various sectors, including healthcare, smart cities, transportation, and agriculture, have begun to use IoT. These applications involve a significant number of sensors and devices that produce a large amount of real-time data. To produce a large amount of data, location and spatial awareness characteristics of devices must be taken into account. After conducting thorough literature, the author finds difficulties while indexing a moving object. According to the author, the major and important challenge is to get the current location of the moving object.

The author [57] works on indexing moving objects in indoor space. According to [57], the majority of people live indoor lifestyles for most of their lives. The positions of moving objects will be an important foundation for many applications, including the tracking of moving objects, wayfinding, and security, with the existing appropriate indoor positioning devices, such as Bluetooth and RFID, WIFI. The author [57] suggests a novel index tree for moving objects in cellular space in this work. In this paper, the author determines the adjacency between the interior environment cells. The interior environment cell will serve as the basis for the Index. In addition to this, the author uses effective query processing to determine the effective updates for moving objects in enclosed spaces.

The author [58], suggests an Internet of Things (IoT) device localization algorithm for indoor environments called the Smartphone-Assisted Localization Algorithm (SALA). With the help of this SALA, a smartphone can visually display where IoT devices are located in indoor spaces, making it simple to manage them for monitoring and remote control. A smartphone functions as a mobile beacon that uses its motion sensors, such as the accelerometer, gyroscope, and magnetometer, to track its location inside buildings. According to the author, the smartphone regularly broadcasts short-range beacon messages while moving about indoors and gathers the response messages from nearby IoT devices. Information about IoT devices is contained in the reply messages. The smartphone stores the location, signal strength, and information about the IoT device in the answer messages into a specific server. These saved trace data are processed offline along with a predetermined indoor layout, such as an apartment layout, by our localization algorithm. It has been demonstrated through simulations that our SALA is capable of accurately localizing IoT devices in an apartment with location errors under 20 cm in a situation that is representative of a real apartment.

An author [59] performs an analysis of the most important indoor technologies and processes to examine their many advantages, drawbacks, and potential areas for development. According to the author, the general metric to evaluate these technologies are scalability, precision, complexity, robustness, energy efficiency, cost, and dependability. The usage of Wi-Fi, RFID, UWB, Bluetooth, ZigBee, and Light over other indoor technologies for dependable, Internet of Things-based applications has been established. The author finds that the complexity of indoor things depends on the unreliability of outdoor systems.

Reference [60] uses machine learning Naïve Bayes to evaluate the performance of the proposed model while tracking and locating the indoor things. The development of internet of things (IoT)-based indoor localization systems is made possible by the widespread use of the internet and the exponential growth in small hardware variety. In this paper, the author calculates the distance between a target and the available reference points using the signals of WiFi or Bluetooth. The target node's position is found by combining the calculated distances. After getting the target nodes, the author uses Naïve Bayes to authenticate the performance of the proposed model whether the proposed model tracks indoor things or not.

Reference [61] proposed the indoor location-based control system to predict the user's indoor position. To locate the indoor position, the localization server, service-provision client, and user application positioning technology are the key components of the system. The service-provision client includes the necessary services, such as interior navigation and monitoring/surveillance. The localization server includes access to terminal devices (such as Smart Phones and other wireless devices). The user application offers the information required for the server to localize the devices or for the user to access a range of client services.

The author [63] introduces the Viewpoint-based Weighted Kernel Fuzzy Clustering (VWKFC) algorithm, which is based on density. To begin with, the author outlines the Kernel-based Hypersphere Density Initialization (KHDI) algorithm as a prerequisite, which replaces the Euclidean distance with the kernel distance and proposes a novel density radius. Secondly, their proposed strategy establishes the concept of weight information granules, which includes a feature weight matrix that assigns different weights to different features to reduce the effect of unrelated features, and a sample weight that is assigned to each data point to reduce the impact of noise and outliers on clustering. Thirdly, the density viewpoint is identified as the data point with the highest local density obtained by KHDI. Their proposed strategy then combines kernel mechanism, density viewpoints, weight information granules, and a maximum entropy regularization to develop the VWKFC algorithm and establish its convergence. The experimental results demonstrate that VWKFC outperforms eight other clustering algorithms on five evaluation indices, particularly when dealing with high-dimensional data.

The author [64] doing a survey on Indoor Positioning Systems for IoT-Based Applications. According to the author, the Internet of Things (IoT) has become an important aspect of both the technology industry and academic research in recent years. This article aims to provide a comprehensive overview of IPSs and localization services in the context of IoT. It first explains the main concepts and reviews the latest positioning methods, techniques, and technologies, with a focus on IoT. It then discusses the technical challenges of implementation and open issues, along with feasible solutions. Finally, the article examines location-based services (LBSs), real IoT applications, and active vendors in the field of positioning services, offering valuable insights into LBSs in IoT for future research.

The author [65] presented IdeAir system, which is an affordable system for monitoring indoor air quality using Internet of Things (IoT) technology. The proposed system is designed to address the limitations of existing alternatives, by detecting and reporting on dangerous gases and their concentrations through alarms and notifications. IdeAir's development process adhered to the Test-Driven Development Methodology for IoT-Based Systems (TDDM4IoTS) and employed an automation tool based on this methodology, simplifying the work of the development team. Early tests indicate that IdeAir has received positive feedback from potential users.

The author [66] introduces a new algorithm that optimizes the positions of UWB radios within a predefined region of interest, while considering the presence of obstacles. To balance the influence of radio geometry and NLOS effects, the optimization problem is formulated using the mean-squared error (MSE) metric. The proposed algorithm is then employed to determine the minimum number of UWB radios necessary for achieving a specific localization accuracy and their respective positions. Experimental results demonstrate that the proposed UWB radio placements reduce the localization root-mean-squared error (RMSE) by 47% and 76% in 2D and 3D experiments, respectively, when compared to trivial placements in real-world cluttered environments.

The author [67] will demonstrate the effectiveness of utilizing the Internet of Things (IoT) through the use of Real-Time Monitoring and I-Beacon technology for indoor healthcare tracking. Due to the rapid and widespread transmission of COVID-19, there is an urgent need for ventilators and personal protective equipment (PPE) to provide healthcare workers and doctors with adequate safety gear. The current Powered Air-Purifying Respirator (PAPR) is not suitable for COVID-19 and social distancing. However, there are limitations to using the Global Positioning System (GPS) in indoor environments such as hospitals and quarantine facilities. Therefore, alternative PAPR tracking methods need to be developed for such settings. Their proposed strategy aims to design a PAPR tracking system that uses Bluetooth Low Energy (BLE) modules as beacons. The hospital will have access to information about the position of the beacons and doctors passing by. When a PAPR, attached to a Bluetooth module, passes by a beacon in the hospital corridors, it will receive position data and send it directly to the ESP32 circuit. The ESP32 will then transmit this information to the control room, allowing for IoT technology to be added to the respirator. This will ensure physical distancing, healthcare staff tracking, and recording.

The author [68] explores the feasibility of using Bluetooth Low Energy (BLE) and Feed Forward Neural Networks (FFNN) for indoor localization applications.

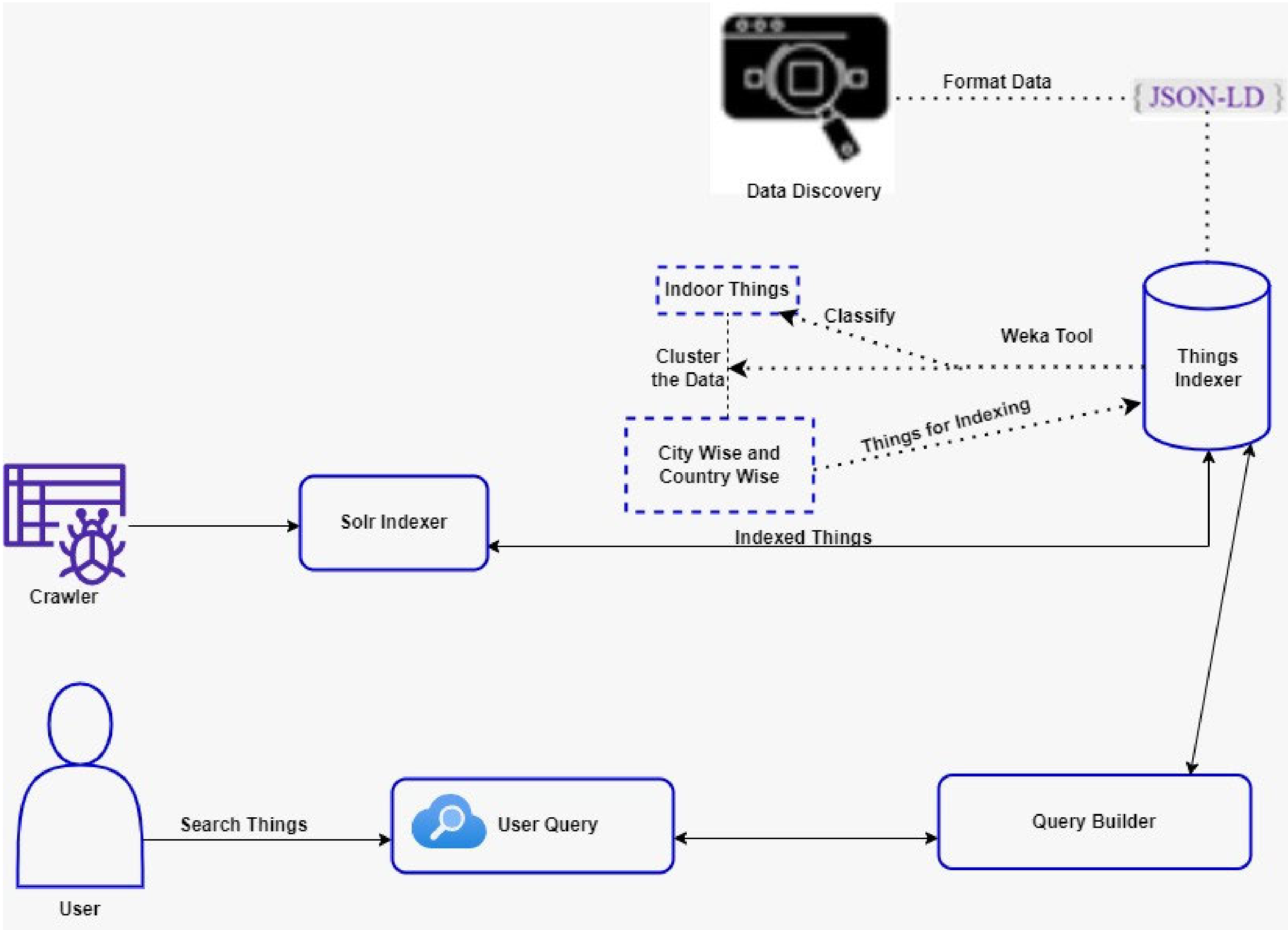



**FIGURE 2.** Proposed indexing architecture.

Thirteen different BLE ibeacon nodes were placed in an indoor environment, and their signal strength values were used to train a FFNN. The FFNN was tested under various hyper-parameter conditions, and it provided reasonably good accuracy of 86% when the batch size was 100 and the learning rate was 0.01. These findings suggest that the FFNN can be implemented in location-based IoT applications.

## III. PROPOSED APPROACH

Figure 2 illustrates the proposed indexing approach.

The brief detail of the steps performed in figure 2 is as follows:

1. In step 1 the proposed approach discover the data by using HTML "Get" and "Post" method. Data is carried out with the help of the "Get" method and passed on to the system with the help of the "Post" method. Once the data is discovered, then the proposed approach passed the data to the format builder.

2. In step 2, the format builder formats the data of Things in JSON-LD format. The reason for formatting data in JSON-LD is that other formats such as WSDL, micro-formats, and DPWS metadata are obsolete and search engines do not understand these formats.

3. Once the data is formatted, we passed the data to the Things indexer. From the Things, indexer data is passed to the Weka tool to perform the classification and clustering of data. We use the Naïve Bayes classifier to classify indoor things and outdoor things. Because in this paper we work on Indoor Things. After the classification of Indoor things, we perform the partitioning cluster approach to group the data city-wise and country-wise. After grouping the data city-wise and country-wise, the data is passed to the Things indexer.

4. Things indexer then passed the data to the Solr Indexer. Before the data was transferred to Solr Indexer, we developed a schema comprised of both static and dynamic properties of Things as depicted in Figure 3.

The next stage is to describe the Description in various formats once the Schema of Things has been established. We need to take a parsing step at this point. Because the Description of Things can be in various JSON, JSON-LD, and XML formats, the parsing stage is crucial. The Things are not indexed in Solr without parsing the Description of the Thing. For parsing, the authors employed the XML parser

**TABLE 3.** Comparison of proposed strategy with the existing approaches.

| Name of Author | Keyword Based Search Engine | Spatial Based Search Engine | IoT Search Engine | WoT Search Engine | Working of Author |
|---|---|---|---|---|---|
| [69] | Yes | No | Yes | NO | They put forward Snoogle as a potential search engine which defines an entity as a group of keywords. However, it may not be suitable for both the dynamic and static nature of devices in a large-scale network. |
| [70] | Yes | No | Yes | No | A system called MAX has been suggested, which relies on tags for detecting entities rather than sensors. MAX necessitates a fast communication speed since it needs to transmit the communication to every tag and substation. |
| [23] | Yes | No | Yes | No | Dyser, a search engine proposal, faces a primary challenge with indexing, as it relies solely on keywords and fails to capture the dynamic characteristics of devices.. |
| [71] | Yes | Yes | Yes | No | IoT-SVK, a search engine that was suggested, conducts searches utilizing both spatial and keyword parameters. However, it is incapable of retrieving data from devices that are dynamic in nature. |
| [27] | No | Yes | Yes | No | A suggestion was made for a search engine that utilizes spatial data obtained from sensors uploaded by their owners. However, the system's limitations include a lack of user-friendly interface and issues with scalability. |
| Proposed Search Engine | Yes | Yes | Yes | Yes | The suggested concept involves a search engine designed to adapt to the dynamic nature of devices. It operates by indexing the devices using their coordinates, as well as keeping track of their historical locations with a location tag. This search engine performs efficient indexing and ranking, with indexing being based on the schema model of the OGC sensor APIs, and ranking based on the type, functionalities, services, and location of the items queried by the user. Notably, these search engines are intended for the Internet of Things, and the proposed one specifically for the Web of Things. |

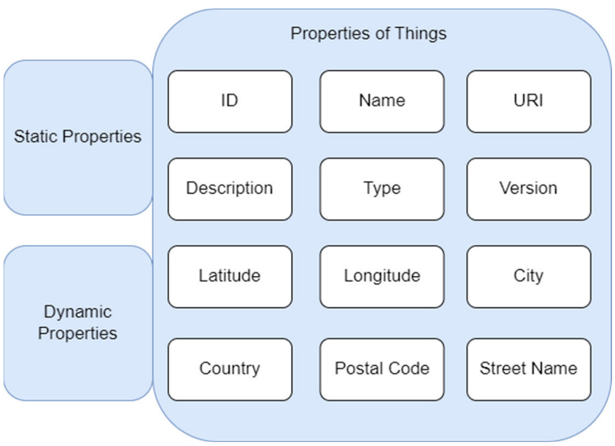



**FIGURE 3.** Properties of things.

and the JSON-LD parser [53], [54]. Other forms, including DPWS metadata, microformats, and WSDL, describe the Description of Things. These formats, which are now out of date, are not understood by search engines or even web browsers. We opt for the JSON, JSON-LD, and XML formats because of this. The Things must then be indexed following the Schema after being parsed.

The data is not indexed if the data type of any element declared in the Schema of Things does not match the element in the Things description file submitted by the authors. The authors of Keywords and Spatial Based Indexing for Searching the Things on the Web execute many indexing techniques. The authors employed a hybrid strategy for indexing, which is the third option. The first is keyword-based indexing, followed by spatial-based indexing. The method of indexing the Description of data based on a few textual words is known as "keyword-based indexing." The Things must then be indexed following the Schema after being parsed.

The authors group the keywords into the name, URI, id, version, type, and Description (services) keywords to index the data to facilitate effective data retrieval. Effective searching is a benefit of categorizing the keywords. To save time, writers have stored the data of Things with location history and current position (latitude and longitude) at the indexing location rather than in the database. When using dynamically updating devices, authors can't record the data in a database because they would have to crawl the real website where the data is stored. If the schema matches the data in the Things indexer only, then the data indexing in Solr indexer will be performed. Otherwise, in case of non-matching of data with the schema the Solr Indexer will not do data indexing. Things were linked with the Solr indexer via COAP protocol.

Indexing in the Web of Things (WoT) is a relatively new area of research that involves organizing and making sense of the massive amounts of data generated by IoT devices. One important aspect of this research is the development of efficient and accurate methods for locating indoor things within smart environments. The major novelty and originality of this research lies in its potential to revolutionize the way we interact with our physical surroundings. By developing efficient and accurate indexing techniques, we can enable a wide range of applications, from smart homes and buildings to industrial automation and healthcare.

One of the main challenges in locating indoor things within a smart environment is the high variability of the physical environment, including changes in lighting, obstacles, and moving objects. To address these challenges, we developed the proposed model that process and analyze the vast amounts of sensor data generated by IoT devices in real-time. In addition to the technical challenges, there are also important ethical and privacy concerns that must be addressed in the development of indexing techniques for the Web of Things in order to improve our quality of life in a wide range of areas. For example, the use of sensor data to track individuals within a smart environment must be done in a way that respects privacy and ensures that personal data is not misused or abused.

To establish the clear superiority of the proposed methodology with the other methods, first of all we have to define the research question and hypothesis.

Research question: How can indexing be used in the Web of Things (WoT) to locate indoor things?

Hypothesis: By developing an indexing system for indoor things in the WoT, it will be possible to efficiently and accurately locate them, leading to improved user experience and increased adoption of the technology.

Once the research question and hypothesis is defined then the next step is to identify the methods used for comparison. The following methods/algorithms are chosen to perform the comparison:

The most commonly used algorithms for indexing in web of things are as: Keyword-based indexing, spatial-based indexing, machine-learning based indexing, and hybrid indexing.

- Keyword-based indexing:

In keyword-based indexing, documents are categorized by a predefined set of keywords, which are used to describe their content. This method is a conventional approach to indexing and involves assigning specific keywords to each document for easy retrieval. While it is relatively easy to implement and provides quick results, keyword-based indexing has its drawbacks as it doesn't account for the contextual meaning of the text.

- Spatial-based indexing:

Spatial-based indexing is a crucial method utilized in the Web of Things (WoT) for arranging and handling physical location-related data. Due to the expansion of the Internet of Things (IoT), the amount of information produced by sensors and gadgets in the real world has significantly increased. In the WoT, spatial-based indexing involves assigning spatial coordinates to each device or sensor, allowing them to be grouped together based on their location using techniques such as Geohashes, Quadkeys, or Spatial Tiles. Indexing devices based on their location makes it easier to search and analyze data by spatial proximity. An instance of the usage of spatial indexing would be a smart city application that organizes data from sensors located throughout the city. By grouping sensors based on their location, the application can effortlessly examine data from a particular neighborhood or street, or identify patterns and trends across the entire city.

- Machine-learning based indexing:

Indexing based on machine learning utilizes algorithms that learn from data to construct an index without human intervention. By training on vast datasets, machine learning algorithms can recognize patterns in the data, which are then leveraged to index documents. As more data is processed, the accuracy of the index continues to improve as the algorithm becomes more proficient at learning from new information.

- Hybrid indexing:

Hybrid indexing is an indexing approach that integrates various indexing techniques, including semantic and keyword-based indexing, to develop a more precise index. By combining the strengths of each method and mitigating their drawbacks, hybrid indexing can enhance search result accuracy while maintaining search speed and efficiency. As a result, hybrid indexing is increasingly gaining popularity.

The next step is to perform the comparison. The table 3 shows the comparison of proposed strategy with the existing approaches.

Once the comparison is performed then the next step is to display the comparison in some graphical form. In this paper we represent the comparison in chart after performing on Tableau a statistical tool in figure 4 at page no 15.

5. After indexing the data in Solr, the proposed approach performs the Crawling step. The process of detecting the resources and the incorporation of their characteristics into indexes is termed crawling. In crawling, a search engine examines a website's content to detect changes and discover new stuff. Traditionally, search engines crawl many websites simultaneously to identify any changes. With WoT, crawling is both manual and automated [47]. When the search engine

crawls manually, it visits the website of each Thing to check for changes to its information. The procedure is repeated every two weeks by several search engines. This leads to an inconsistent output for the user. A process known as automatic crawling involves the search engine detecting new information about Things either via autonomous agents or by registering them at the indexing location. Resource discovery should be centralized when crawling is automatic [47] so that the search engine can receive the most current material in real-time.

In this approach, we perform automatic crawling on Solr Indexer so that if any change in data occurs it will automatically change in Solr Indexer also.

6. As presented in Figure 2, it is indicated that we have developed a user interface that receives a query from the user and then transfers it to the query builder, where this query will be processed. At this stage, it will also be found whether the user searches via keyword-based query or spatial-based query or simply say that whether the user searches through static or dynamic properties. The proposed approach supports hybrid indexing. After this step, the query builder will transfer this query to Things Indexer where this query will be matched with the data present there if the matching is successful then it passes to Solr Indexer, carries the data, and returns it to the query builder and then to the user interface. In this way, the proposed indexing architecture helps the user in finding Things.

There are many theoretical advantages of the proposed strategy such as:

- Improved efficiency: Indexing indoor things in the WoT can improve the efficiency of locating and accessing them. This can be particularly useful in large indoor environments, such as airports or hospitals, where finding a specific object or device can be challenging and time-consuming.
- Enhanced accessibility: By making devices and objects more easily discoverable, indexing can enable people with disabilities to use them more easily. For example, indexing can help people with visual impairments find and use devices with audio output, such as speakers or voice assistants.
- Energy Efficient: By indexing devices, it becomes possible to automate routine tasks, such as turning off the lights when leaving a room or adjusting the temperature based on occupancy. This can improve energy efficiency and reduce the workload on users.

Enable better analytics: Indexing can enable better analytics by providing more comprehensive data about indoor things. By indexing devices and objects, it becomes possible to track their usage and interactions more accurately. This can provide valuable insights into patterns of behavior and usage that can inform the design of better indoor environments.

We have employed a Mathematical model to explain/describe the proposed WoT Search engine [39].

### A. MATHEMATICAL MODEL OF PROPOSED WOT SEARCH ENGINE

We have developed a mathematical model. As Finding the important elements and how they interact with one another inside the system is necessary for a mathematical model so we have described the process in steps. The steps which we take to develop a mathematical model of the proposed WoT search engine are as follows:

- What are the inputs and outputs? Finding the inputs and outputs of the WoT search engine is the first stage. The search query is the input, and the list of things that match the query is the output. In our case, the input data is to get the data of Things from sensorThings API.
- Define the search algorithm: The set of rules used to match the search query with the data of various items is known as the search algorithm. The type, name, URI, and location of the Things are the basis for the search algorithm.
- Define the things indexing: All the items that can be searched for are included in the things indexing. It contains details on the thing's type, name, features, and any further pertinent information.
- Describe the performance measurements: The WoT search engine's performance is assessed using the performance metrics. The measures could include MCC, F1 score, recall, and precision.

After analyzing the system model, we used some variables, including the quantity of data, the speed of the search, and the effectiveness of the algorithm used to create the model. Let's define some variables that will help us build the model:

N: the number of Things

D: the amount of data generated by each Thing

T: the time required to search for data

S: the search speed of the WoT search engine

E: the efficiency of the algorithm used by the WoT search engine

We can use these variables to create an equation that describes the performance of the WoT search engine:

Data Searched per second = N * D * E / (T * S)

According to this equation, the number of Things, the volume of data produced, and the algorithm's effectiveness are all directly proportional to the amount of data that can be searched per second, while the time needed to search for data and the speed of the WoT search engine are inversely related.

Figure 4, shows how these variables relate to one another. The amount of data that can be searched per second will be represented on the y-axis of a line graph, with the x-axis representing the search speed of the WoT search engine.

### B. MATHEMATICAL MODEL OF CRAWLING STEP IN WOT SEARCH ENGINE

The process of crawling entails creating a set of equations that explain how the search engine will behave when it explores the Web of Things (WoT) and gathers information.

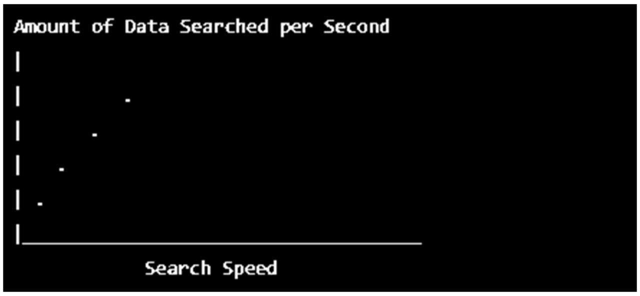



**FIGURE 4.** Represent how variables relate to one another.

N: The total number of Things.

M: The number of devices visited by the search engine at time t.

p: The probability of the search engine selecting a thing to visit at time t.

Q: The total number of data points collected by the search engine.

R: The rate at which the search engine collects data.

T: The total time spent crawling the WoT.

Based on these variables, we construct the following equations:

M(t) = N * (1 - (1 - p)^t) – This equation calculates the number of things the search engine has visited at time t, assuming a uniform distribution of things selection.

Q(t) = M(t) * R * t – This equation calculates the total number of data points collected by the search engine at time t, based on the number of things visited and the rate of data collection.

T = N/p – This equation calculates the total time required to visit all things in the WoT, assuming that each thing is visited once.

### C. MATHEMATICAL MODEL OF INDEXING STEP IN WOT SEARCH ENGINE

There are multiple processes in a mathematical model for WoT search engine that categorizes indoor sensors:

- Specify the input parameters: The sensor data from the WoT search engine, such as temperature, humidity, light level, and sound level, might be one of the input factors. Symbols such as T for temperature, H for humidity, L for light level, and S for sound level are used to denote these factors.
- Identify the output parameters: The classification of the sensors based on the input data is one of the output variables. The sensors might fall within the categories of temperature sensors, humidity sensors, light sensors, or sound sensors.
- Develop the model: The model was developed using a naïve Bayes or a neural network.
- Validate the model: A dataset of sensor readings and the accompanying classifications can be used to validate the model. By contrasting the projected classifications with the actual classifications, the model's accuracy may be evaluated.

A straightforward mathematical classification scheme for indoor temperature sensors.

If T > 25, then classify as "hot sensor" If 20 < T <= 25, then classify as "warm sensor" If 15 < T <= 20, then classify as "normal sensor" If T <= 15, then classify as "cold sensor"

Figure 5 shows how to classify temperature sensors based on their temperature readings.

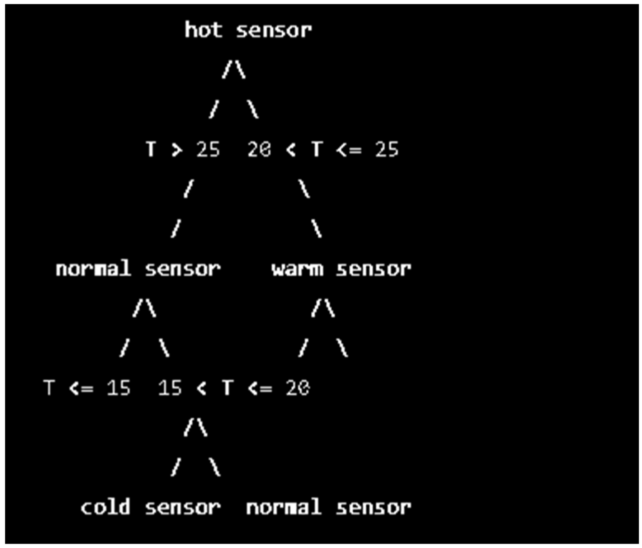



**FIGURE 5.** Represent how to classify temperature sensors based on their temperature readings.

In Figure 5, the temperature variable T is used to classify the sensors into different categories based on their temperature readings. Figure 5 shows the decision points and the resulting classifications for each range of temperature values.

### D. A MATHEMATICAL MODEL OF WOT SEARCH ENGINE CLASSIFYING THE SENSORS AS COUNTRY WISE AND CITY

Various processes in a mathematical model for the WoT search engine categorize sensors as city- and country-wise:

- Specify the input parameters: The sensor data and location data are among the input factors for the WoT search engine. Any form of sensor readout, including those for temperature, humidity, light level, or sound level, could be included in the sensor data. The latitude and longitude of the sensor may be included in the location information.
- Identify the output parameters: The classification of the sensors based on the position information is one of the output variables. The sensors might be categorized as being in a certain nation or city. A sensor might be categorized as being in the United States or the city of New York, for instance.
- Creating the model: One method for creating the model is partition clustering. Based on the position information of the sensors, this algorithm divides them into various clusters. Each cluster represents a city or country.

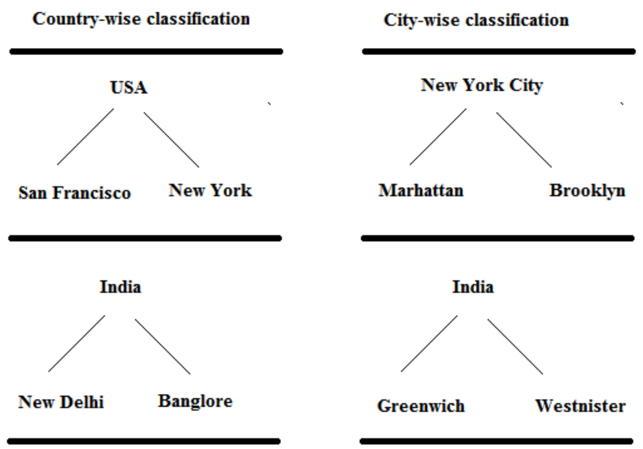



**FIGURE 6.** Represent how the partition clustering model would classify sensors based on their location data.

- Validate the model: A dataset of sensor readings and their related locations can be used to validate the model. By contrasting the projected categories with the actual locations, the model's accuracy may be evaluated.

A straightforward mathematical model that uses partition clustering to categorize sensors into rural and urban areas could be:

Let C be the set of all countries and S be the set of all sensors to classify countries. Let (lat, long) be the latitude and longitude of each sensor s in S. The country c that s belongs to is given by:

c = argmin d((lat, long), cc)

where cc is the centroid of the cluster for country c, and d is a distance function, such as the Euclidean distance.

Let C be the set of all cities and S be the set of all sensors to classify cities. Let (lat, long) be the latitude and longitude of each sensor s in S. The city c that s is a part of is identified by:

c = argmin d((lat, long), cc)

where cc is the centroid of the cluster for city c, and d is a distance function, such as the Euclidean distance.

Figure 6 shows how the partition clustering model would classify sensors based on their location data:

Based on their location information, the sensors in this model are grouped into various clusters. Depending on the classification objective, each cluster corresponds to a country or a city. The country and city clusters, as well as the related sub-clusters, are displayed on the graph.

## IV. EVALUATION

This section explains the study questions, data set, metrics, and evaluation procedure for the suggested approach.

### A. RESEARCH QUESTIONS

The proposed strategy is assessed by examining the following questions of interest:

Q1: Investigation and comparison of the proposed indexing approach with the existing indexing approach for the web of things.

Q2: Which clustering technique is best for our data set?

Q3: How to identify the Things as indoor and outdoor things?

Q4: Whether the proposed algorithm is better as compared to the algorithms discussed in the state-of-the-art?

Q5: Whether proposed indexing approach gives accurate results or not by using machine learning?

The first research question investigates the proposed indexing approach with the previous indexing approaches as mentioned in the state-of-art section.

The second research question analyzes the accuracy of the proposed indexing approach by using the machine learning tool Weka.

### B. DATASET

The dataset is compiled using Energy Sensor data Environment sensor data from Sensor Web APIs, and W3C sensors API. The proposed model fetches sensors' data from the Sensor Web APIs and uses the "RESTFUL APIs." The RESTFUL API consists of HTTP and identifiers (URL). With the help of the HTML gets function, the proposed model retrieves the current data of the sensor. With the help of the post function, the proposed model retrieves the changes in data. The fetched data is formatted into three different formats XML, JSON, and JSON-LD. Once the Things data are formatted, it is passed to the Things indexer which matches the data according to the schema. After indexing the data, we convert the data into CSV format by labeling the country. We give a label 1 to Pakistan, 2 to the USA, 3 to Canada, 4 to Germany, and 5 to Japan.

### C. METRICS

The accuracy, precision, recall, and f-measure, which are the most common and well-known metrics, are used to measure the performance of the proposed method. The equations for these metrics are shown below [30].

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F - Measure = \frac{2 * Precision * Recall}{Precision + \mathrm{Recall}}$$

$$MCC = \frac{TP * TN - FP * FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

The authors calculated the accuracy by using the True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) values. The output consists of positive and negative elements, which depend on TP, TN, FP, and FN.

1. True Positive: The output is true positive if the search engine displays the results exactly according to the query.

2. True Negative: The output is true negative if the search engine displays results other than the exact query.

3. False Positive: The output is false positive if the search engine incorrectly identifies the accurate results.

4. False Negative: The output is false negative if the search engine incorrectly identifies the inaccurate results.

### D. PROCESS

We have uploaded the description of 200 sensors in Windows 7, using a 32-bit operating system. Results were taken concerning accuracy, precision, recall, and f-measure.

### E. RESULTS AND DISCUSSION

***Research Question 1: Investigation and comparison of the proposed indexing approach with existing indexing approaches for the web of things.***

To investigate research question 1, we compared the proposed indexing approach and existing indexing approach engines based on three different parameters as Keyword-based indexing, spatial indexing, and hybrid indexing. The proposed approach uses hybrid indexing as discussed in the proposed approach section.

We have used Tableau a statistical tool [48] to analyze the performance of the proposed indexing approach and compare it with the existing indexing approaches. To do this we assign different values according to the literature. We assign a value of 0-6 to keyword-based indexing. As keyword-based indexing contains 6 different parameters such as ID, Name, URI, Type, Description, and Version. The approach which does not use any keyword assigns a value of 0. The approaches use a single keyword to assign a value of 1, which uses two keywords to assign a value of 2 and which uses three keywords to assign a value of 3, and so on. We assign a value of 0 to 2 to spatial-based indexing. We assign a value of 0, which does not use any spatial indexing. As spatial-based indexing contains coordinates and relative location. We assign a value of 1 if the approach uses single spatial indexing and assign a value of 2 if the approach uses multiple spatial-based indexing. We assign a value of 2 which uses a hybrid-based indexing approach. After analyzing this on Tableau, the proposed indexing approach performs better in comparison with other indexing approaches as shown in Fig. 4.

***Research Question 2: Which clustering technique is best for our data set?***

To investigate research question 2, we used the Weka tool to cluster the data by using partitioning clustering. We choose partitioning cluster because we want to cluster the data into two groups' indoor and outdoor things by using similarities and dissimilarities. Other clustering techniques such as hierarchical clustering, grid clustering, density clustering, and centroid clustering do not cluster the data based on similarities and dissimilarities. For instance, Hierarchical clustering is best for the data if we want to cluster the data in tree form or taxonomy. Density-based clustering is best for the data if we want to cluster the data according to the shape of the data. Grid-based clustering is best for the data if we want to cluster the data according to the space required by data in memory. Centroid-based clustering is best for the data if we want to cluster the data according to the physical location of the data. In our case, we simply want to divide our data into two groups. For this kind of data clustering, a partitioning cluster is the best approach.

***Research Question 3: How to identify the Things as indoor and outdoor things?***

To investigate research question 3, we used the services, and description of Things to identify whether the Thing is an indoor or outdoor thing. As we bring data through APIs thus we can estimate whether the Things are indoor or outdoor by the use of its description or services. For instance, if it is written in Description of Things that the Thing will calculate the distance between buildings then it will be outdoor while if the distance between objects present at the same location is to be calculated then it would be indoor. Similarly, if a Thing calculates a raindrop it is outdoor and if it calculates the temperature of a room, it is indoor. We identify such types of Things and transfer them to the Weka tool, which splits our data into two groups giving us separate data on Indoor and outdoor Things.

***Research Question 4: Whether the proposed algorithm is better as compared to the algorithms discussed in the state-of-the-art?***

To investigate research question 4, we compared the working and drawbacks of the existing algorithms discussed in the state-of-the-art section with the proposed algorithm. Table 4 represents the work of the authors along with the drawback of their proposed algorithm.

***Research Question 5: Whether the proposed indexing approach gives accurate results or not by using machine learning?***

To investigate research question 5, we used the Weka tool to validate the accuracy of the proposed indexing approach.

The reason for using the Machine learning-based method instead of traditional methods is because the nature of the data we used is continuous. With the continuous nature of data, machine learning models are best to use. Traditional method includes rule-based learning. Rule-based learning is performed by setting the rules using if-else. In rule-based learning, if we have to perform 50 different actions then we have to make 50 different rules. Suppose we perform Rule-based learning to validate the results of our proposed model. Then we have to create two Rules to identify whether it is indoor things or outdoor things. Rule 1 contains all the descriptions and services of indoor things. Rule 2 contains all the descriptions and services of Outdoor things. After

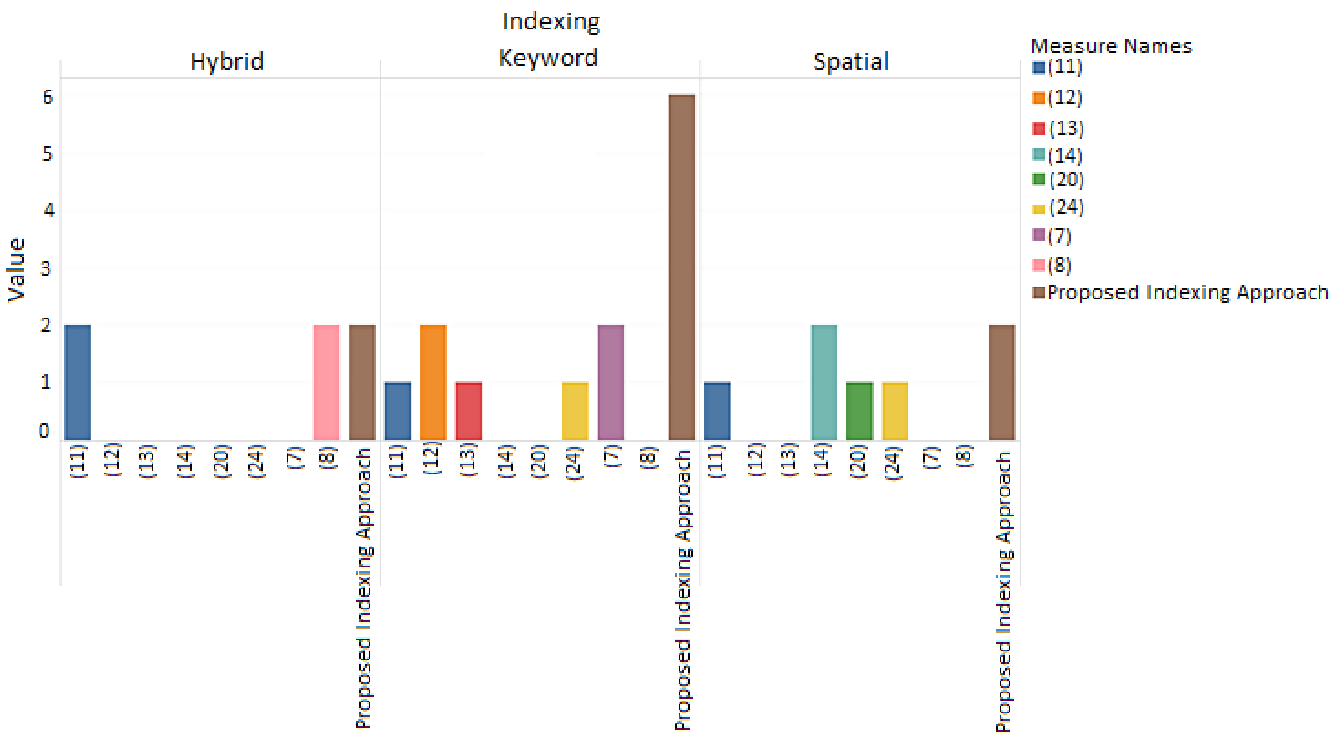



**FIGURE 7. Comparison of existing indexing approaches and proposed indexing approach using tableau.**

**TABLE 4. Working of authors.**

| Author | Clustering | | | | | Identify Things | | | Evaluating Accuracy | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Partition | Density | Grid | Hierarchical | K-means | Description | Distance | Functionality/Services | Using Traditional Method | Using Machine Learning |
| [39] | | | | | ✓ | | | | ✓ | |
| [40] | | | | | ✓ | | | | ✓ | |
| [41] | ✓ | | | | | ✓ | | | | |
| [43] | | | | ✓ | | ✓ | | | | |
| [44] | | ✓ | | | | ✓ | | ✓ | | |
| [46] | | | | | ✓ | | ✓ | | | |
| [55] | | | ✓ | | | | ✓ | | | |
| [59] | | | | | | | ✓ | | | |
| [60] | | | | | | | ✓ | | | ✓ |
| Proposed Algorithm | ✓ | | | | | ✓ | ✓ | ✓ | | ✓ |

identification of Indoor and Outdoor, we make two different rules to identify the city and country. Similarly, rule 3 is to identify the city, and rule 3 is to identify the country. Then we have to make a final rule to validate the results of the proposed model. So in short we have to make 5 different rules. But in 5 different rules, we have to perform nested if-else as shown below in the algorithm, pseudo-code, and figure 5 [62].

Algorithm:

1. Step 1. Defined a rule1 which contains the description of indoor things in it.

2. Step 2. Defined a rule 2 which contains the description of outdoor things in else.

3. Step 3. In the if of step 1 we further used if-else to identify the city.

4. Step 4. In the if of step 1 we further used if-else to identify the country.

5. Step 5. Defined a final rule to authenticate the result's validity of the proposed model.

Pseudo-code:

Input data from sensor APIs

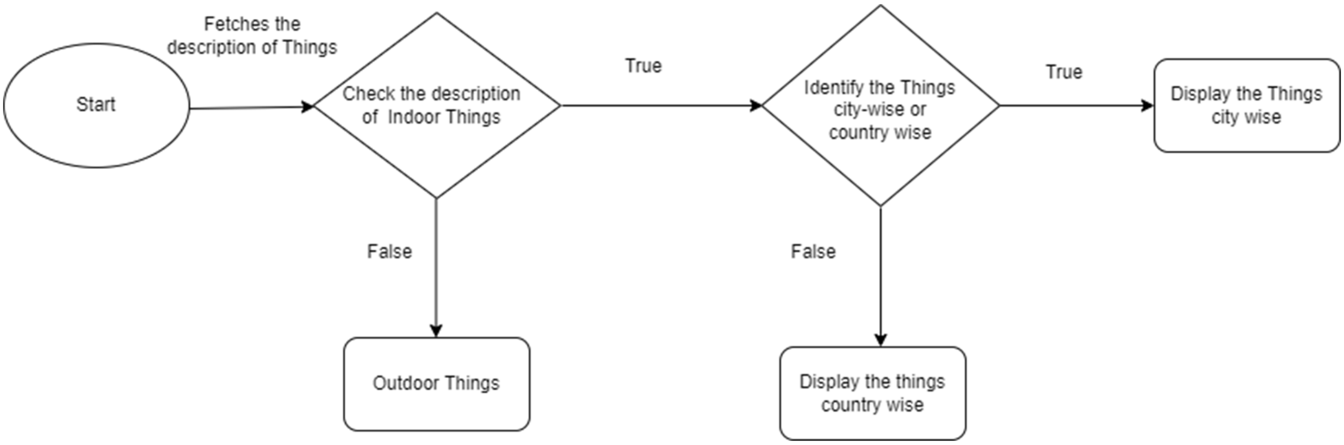



**FIGURE 8.** Block Diagram to represent the algorithm.

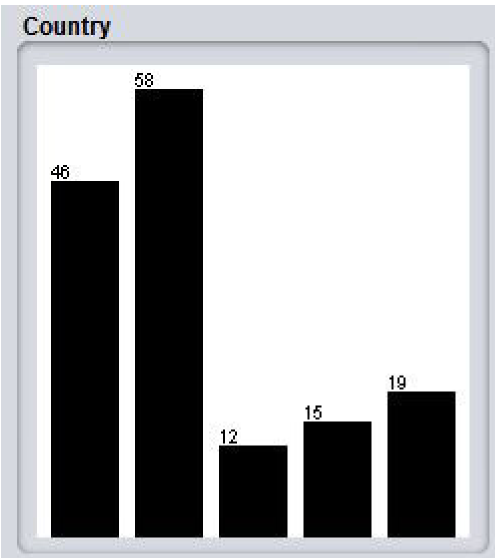



**FIGURE 9.** Classification of country from training data set.

**If** des = for
**If** Schema = D indexed_data($S_{index}$)
True
**If** Th= CW
True
Display Result
**else**
Display Result ContW
else
Display Outdoor Things

As seen in the Algorithm of rule-based learning, we do not authenticate the accuracy or validity of results as rules are defined by the researcher. Below are the advantages of rule-based learning and machine learning-based methods.

Rule-based learning VS ML-based method [62]

• In machine learning, a lot of labeled data has to be fed into computers as the training data to develop a Machine Learning model with high accuracy. On the other hand, in the rule-based approach, you have to hardcode every single detail but the data required is quite light.

• Machine Learning involves complex mathematical operations, so the Machine Learning approach requires specialized hardware such as GPUs for optimized training. In rule-based systems, training and prediction can be carried out on the same machine without any specialized hardware.

• It is easier to debug the rule-based approach as it requires tedious programming. Machine Learning models, on the other side, require straightforward training but are difficult to debug because it was the computer that came up with the rules and patterns.

• Machine Learning models can constantly adapt and evolve with the continuous streams of data and, therefore, enhance their accuracy over time. However, the intelligence of rule-based systems is limited.

So in our case, the nature of data is continuous, that's why we prefer Machine learning models over the rule-based learning method.

In the Weka tool, we used the Naïve Bayes classifier to validate the proposed indexing approach. To do this, we have to make a data set of 200 Things. Out of 200, we make 150 for the training data set and 50 for the testing data set. To check the validity of the proposed approach we focus on the results of a search by city and country. We choose 5 countries and 5 cities to check the validity, as shown in figure 6 and figure 7 taken from the Weka tool after the training data set is given to it.

In Figure 6, after applying Naïve Bayes on the Weka tool on the training data set, 46 sensors are located in the USA, 58 sensors are located in Pakistan, 12 sensors are located in Germany, 15 sensors are located in Canada and 19 sensors are located in Japan. Then we give the test data set to the Weka tool on the training data set.

In the test data set 10 sensors are located in the USA, 20 sensors are located in Pakistan, 6 sensors are located in

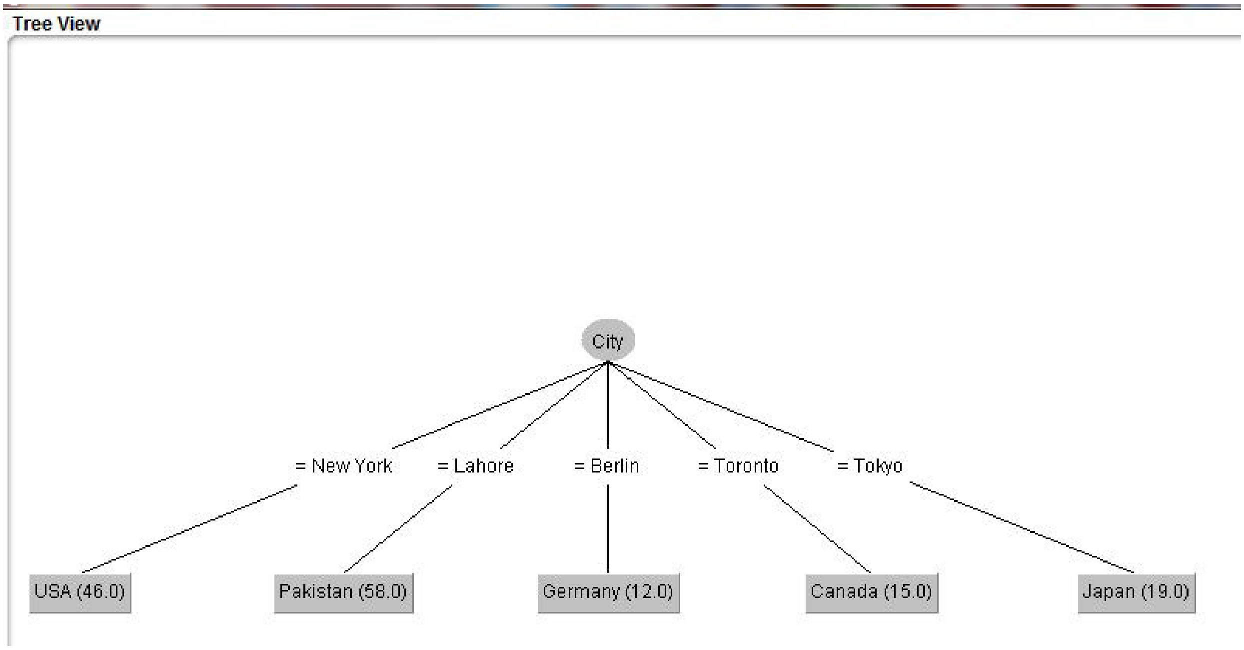



**FIGURE 10.** Tree view of countries and city taken from Weka tool after training.

**TABLE 5.** Training data set while using both classifiers.

| | Classification of City wise using Neural Network | Classification of City wise using Naïve Bayes | Classification of Country wise using Neural Network | Classification of City wise using Naïve Bayes |
|---|---|---|---|---|
| CCI | 150 100% | 150 100% | 150 100 % | 150 100% |
| ICI | 0 0 | 0 0 | 0 0 | 0 0 |
| KS | 1 | 1 | 1 | 1 |
| MAE | 0.0048 | 0.0001 | 0.0048 | 0.0001 |
| RMSE | 0.0072 | 0.0004 | 0.0072 | 0.0004 |
| RAE | 1.667% | 0.0199% | 1.667% | 0.0199% |
| RRSE | 1.8894% | 0.1025% | 1.8894% | 0.1025% |
| TNI | 150 | 150 | 150 | 150 |

**TABLE 6.** Testing data set while using both classifiers.

| | Classification of City wise using Neural Network | Classification of City wise using Naïve Bayes | Classification of Country wise using Neural Network | Classification of Country wise using Naïve Bayes |
|---|---|---|---|---|
| CCI | 48 97.9592 % | 48 97.9592 % | 48 97.9592 % | 48 97.9592 % |
| ICI | 1 2.0408 % | 1 2.0408 % | 1 2.0408 % | 1 2.0408 % |
| KS | 0.9724 | 0.9724 | 0.9724 | 0.9724 |
| MAE | 0.0122 | 0.0081 | 0.0134 | 0.006 |
| RMSE | 0.0759 | 0.0892 | 0.0895 | 0.0657 |
| RAE | 4.1076% | 2.7336% | 4.5245% | 2.0194% |
| RRSE | 19.5642% | 22.9878% | 23.0673% | 16.9507% |
| TNI | 49 | 49 | 49 | 49 |

Germany, 8 sensors are located in Canada and 5 sensors are located in Japan. Following is the confusion matrix on the training and test data set.

The learning rate is used to train a model in identifying a minimal loss when using a neural network. We perform the Neural Network and Naïve Bayes of the training data set

**TABLE 7.** Accuracy of proposed search engine after evaluation from weka using neural network.

| Country | TP | TN | FP | FN | Accuracy |
|---|---|---|---|---|---|
| Pakistan | 52 sensors are displayed out of 58 sensors. | 9 sensors are displayed which are not in Pakistan | 6 sensors are incorrectly identified. | 2 sensors are incorrectly identifies which is inaccurate. | 0.88 |
| USA | 38 sensors are displayed out of 46. | 7 sensors are displayed which are not in the USA. | 8 sensors are incorrectly identified. | 2 sensors are incorrectly identifies which is inaccurate. | 0.81 |
| Germany | 8 sensors are displayed out of 12. | 5 sensors are displayed which are not in Germany. | 5 sensors are incorrectly identified. | 3 sensors are incorrectly identifies which is inaccurate. | 0.61 |
| Canada | 12 sensors are displayed out of 15. | 7 sensors are displayed which are not in Canada. | 4 sensors are incorrectly identified. | 4 sensors are incorrectly identifies which is inaccurate. | 0.67 |
| Japan | 14 sensors are displayed out of 19. | 5 sensors are displayed which are not in Japan. | 3 sensors are incorrectly identified. | 2 sensors are incorrectly identifies which is inaccurate. | 0.79 |
| Total | 124 | 33 | 26 | 13 | 0.80 |

**TABLE 8.** Accuracy of proposed search engine after evaluation from weka using Naïve bayes.

| Country | TP | TN | FP | FN | Accuracy |
|---|---|---|---|---|---|
| Pakistan | 52 sensors are displayed out of 58 sensors. | 9 sensors are displayed which are not in Pakistan | 6 sensors are incorrectly identified. | 2 sensors are incorrectly identifies which is inaccurate. | 0.88 |
| USA | 38 sensors are displayed out of 46. | 7 sensors are displayed which are not in the USA. | 8 sensors are incorrectly identified. | 2 sensors are incorrectly identifies which is inaccurate. | 0.81 |
| Germany | 8 sensors are displayed out of 12. | 5 sensors are displayed which are not in Germany. | 5 sensors are incorrectly identified. | 3 sensors are incorrectly identifies which is inaccurate. | 0.61 |
| Canada | 12 sensors are displayed out of 15. | 7 sensors are displayed which are not in Canada. | 4 sensors are incorrectly identified. | 4 sensors are incorrectly identifies which is inaccurate. | 0.67 |
| Japan | 14 sensors are displayed out of 19. | 5 sensors are displayed which are not in Japan. | 3 sensors are incorrectly identified. | 2 sensors are incorrectly identifies which is inaccurate. | 0.79 |
| Total | 124 | 33 | 26 | 13 | 0.80 |

as well as on the test data set. After analyzing the results of Neural Network and Naïve Bayes, the accuracy is the same. Table 4, Table 5, Table 6, and Table 7 represents the results of both classifiers.

So after using the Machine learning tool Weka we validate that the proposed approach gives 80% accurate results. The Table8 shows the F-measure, recall, precision, and MCC in each case as well as the overall.

**TABLE 9. F-Measure, precision, and recall after evaluation from Weka.**

| Country | F-Measure | Precision | Recall | MCC |
|---|---|---|---|---|
| Pakistan | 0.92 | 0.89 | 0.96 | 0.63 |
| USA | 0.88 | 0.82 | 0.95 | 0.50 |
| Germany | 0.66 | 0.61 | 0.72 | 0.23 |
| Canada | 0.75 | 0.75 | 0.75 | 0.38 |
| Japan | 0.84 | 0.82 | 0.875 | 0.48 |
| Total | 0.85 | 0.82 | 0.90 | 0.50 |

## V. CONCLUSION

The demand for WoT search engines is increasing day by day due to the rapid increase in the usage of Sensors in real-world life. To perform efficient searching, the search engine performs indexing. Search engines are of two types, one is used to search for general data, and the other is used to search for sensor data. There are two crucial components of both types of search engines: one is indexing, and the other is ranking. Without indexing, the search engine is not capable of storing data efficiently.

In this paper, we have focused on searching indoor things by using machine learning tools and algorithms. We used the Weka tool to cluster the data by grouping it into indoor things and outdoor things. Once the data is grouped, and then we used Naïve Bayes classifier to classify the Things city-wise and country-wise. Then we analyzed the results whether our proposed search engine gives accurate results or not. So the accuracy of results given by the proposed search engine is 80%.

The vision of future work in this domain involves the development of more efficient and accurate indexing algorithms and techniques. These techniques will enable users to locate, interact, and integrate the indoor things into WoT more easily. Some potential areas of focus for future work in this domain include:

- Semantic indexing: The development of algorithms and techniques that can understand the meaning and context of indoor things. This will allow for more accurate indexing and search results, as well as more intelligent and personalized recommendations.
- Integration with IoT sensors: The integration of indoor sensors, such as cameras and microphones, into the indexing process. This will allow for more accurate and real-time indexing, as well as more personalized and context-aware results.
- Privacy and security: The development of techniques that can ensure the privacy and security of indoor things, while still allowing for efficient indexing and search.

Overall, the vision of future work in this domain involves the development of more intelligent, personalized, and context-aware indexing algorithms that can seamlessly integrate indoor things into the WoT.

## REFERENCES


[1] S. Kumar, P. Tiwari, and M. Zymbler, "Internet of Things is a revolutionary approach for future technology enhancement: A review," *J. Big Data*, vol. 6, no. 1, pp. 1–21, Dec. 2019.

[2] *Ericsson Technology Review: New Technology, Open Discussions*. Ericsson Technology Review. Accessed: Oct. 15, 2022. [Online]. Available: https://www.ericsson.com/en/reports-and-papers/ericsson-technology-review

[3] H. Liu, H. Darabi, P. Banerjee, and J. Liu, "Survey of wireless indoor positioning techniques and systems," *IEEE Trans. Syst., Man Cybern., C, Appl. Rev.*, vol. 37, no. 6, pp. 1067–1080, Nov. 2007.

[4] A. K. M. M. Hossain and W.-S. Soh, "A survey of calibration-free indoor positioning systems," *Comput. Commun.*, vol. 66, pp. 1–13, Jul. 2015.

[5] Y. T. Chan and K. C. Ho, "A simple and efficient estimator for hyperbolic location," *IEEE Trans. Signal Process.*, vol. 42, no. 8, pp. 1905–1915, Aug. 1994.

[6] M. Husen and S. Lee, "Indoor location sensing with invariant Wi-Fi received signal strength fingerprinting," *Sensors*, vol. 16, no. 11, p. 1898, Nov. 2016.

[7] G. Shen, "Walkie-Markie: Indoor pathway mapping made easy," in *Proc. USENIX Symp. Netw. Syst. Des. Implement.*, Lombard, IL, USA, Apr. 2013, pp. 85–98.

[8] C. Wu, Z. Yang, Y. Liu, and W. Xi, "WILL: Wireless indoor localization without site survey," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 4, pp. 839–848, Apr. 2013.

[9] A. Narzullaev and Y. Park, "Novel calibration algorithm for received signal strength based indoor real-time locating systems," *AEU Int. J. Electron. Commun.*, vol. 67, no. 7, pp. 637–644, Jul. 2013.

[10] M. Ali, S. Hur, and Y. Park, "LOCALI: Calibration-free systematic localization approach for indoor positioning," *Sensors*, vol. 17, no. 6, p. 1213, May 2017.

[11] G. Minaev, A. Visa, and R. Piché, "Comprehensive survey of similarity measures for ranked based location fingerprinting algorithm," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat. (IPIN)*, Sep. 2017, pp. 1–4.

[12] J. Torres-Sospedra, P. Richter, G. Mendoza-Silva, E. S. Lohan, and J. Huerta, "Characterising the alteration in the AP distribution with the RSS distance and the position estimates," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat. (IPIN)*, Sep. 2018, pp. 1–8.

[13] M. R. Faheem, T. Anees, and M. Hussain, "The web of things: Findability taxonomy and challenges," *IEEE Access*, vol. 7, pp. 185028–185041, 2019.

[14] Y. Fathy, P. Barnaghi, and R. Tafazolli, "Large-scale indexing, discovery, and ranking for the Internet of Things (IoT)," *ACM Comput. Surv.*, vol. 51, no. 2, pp. 1–53, Mar. 2019.

[15] Y. Zhou, S. De, W. Wang, and K. Moessner, "Search techniques for the web of things: A taxonomy and survey," *Sensors*, vol. 16, no. 5, p. 600, Apr. 2016.

[16] P. Barnaghi, W. Wang, L. Dong, and C. Wang, "A linked-data model for semantic sensor streams," in *Proc. IEEE Int. Conf. Green Comput. Commun. IEEE Internet Things IEEE Cyber, Phys. Social Comput.*, Aug. 2013, pp. 468–475.

[17] L. Zhang, D. Jeong, and S. Lee, "Data quality management in the Internet of Things," *Sensors*, vol. 21, no. 17, pp. 1–21, 2021, doi: 10.3390/s21175834.

[18] A. Klein, H.-H. Do, G. Hackenbroich, M. Karnstedt, and W. Lehner, "Representing data quality for streaming and static data," in *Proc. IEEE 23rd Int. Conf. Data Eng. Workshop*, Apr. 2007, pp. 3–10.

[19] A. Klein and W. Lehner, "Representing data quality in sensor data streaming environments," *J. Data Inf. Qual.*, vol. 1, no. 2, pp. 1–28, Sep. 2009.

[20] T.-A. Hoang-Vu, H. T. Vo, and J. Freire, "A unified index for spatio-temporal keyword queries," in *Proc. 25th ACM Int. Conf. Inf. Knowl. Manage.*, Oct. 2016, pp. 135–144.

[21] A. Khodaei, C. Shahabi, and A. Khodaei, "Temporal-textual retrieval: Time and keyword search in web documents," *Int. J. Next-Gener. Comput.*, vol. 3, no. 3, pp. 288–312, 2012.

[22] S. Nepomnyachiy, B. Gelley, W. Jiang, and T. Minkus, "What, where, and when: Keyword search with spatio-temporal ranges," in *Proc. 8th Workshop Geographic Inf. Retr.*, Nov. 2014, pp. 1–8.

[23] B. Ostermaier, K. Romer, F. Mattern, M. Fahrmair, and W. Kellerer, "A real-time search engine for the web of things," in *Proc. Internet Things (IoT)*, Nov. 2010, pp. 1–8.

[24] I. Nadim, Y. Elghayam, and A. Sadiq, "Semantic discovery architecture for dynamic environments of web of things," in *Proc. Int. Conf. Adv. Commun. Technol. Netw. (CommNet)*, Apr. 2018, pp. 1–6.

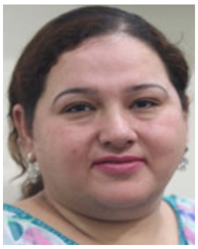
[25] Z. Li, K. C. K. Lee, B. Zheng, W.-C. Lee, D. Lee, and X. Wang, "IR-tree: An efficient index for geographic document search," *IEEE Trans. Knowl. Data Eng.*, vol. 23, no. 4, pp. 585–599, Apr. 2011.

[26] M. Zhou and Y. Ma, "A web service discovery computational method for IoT system," in *Proc. IEEE 2nd Int. Conf. Cloud Comput. Intell. Syst.*, Oct. 2012, pp. 1009–1012.
[27] A. Kansal, S. Nath, J. Liu, and F. Zhao, "SenseWeb: An infrastructure for shared sensing," *IEEE MultimediaMag.*, vol. 14, no. 4, pp. 8–13, Oct. 2007.
[28] A. U. Rehman, M. Hussain, M. Idress, A. Munawar, M. Attique, F. Anwar, and M. Ahmad, "E-cultivation using the IoT with adafruit cloud," *Int. J. Adv. Appl. Sci.*, vol. 7, no. 9, pp. 75–82, Sep. 2020.
[29] H. Butler, M. Daly, A. Doyle, S. Gillies, S. Hagen, and T. Schaub, *The Geojson Format*, Internet Engineering Task Force (IETF), Fremont, CA, USA, 2016.
[30] L. Sciullo, L. Gigli, A. Trotta, and M. D. Felice, "WoT store: Managing resources and applications on the web of things," *Internet Things*, vol. 9, Mar. 2020, Art. no. 100164.
[31] A. Ciortea, S. Mayer, S. Bienz, F. Gandon, and O. Corby, "Autonomous search in a social and ubiquitous web," *Pers. Ubiquitous Comput.*, vol. 24, pp. 1–14, Jun. 2020.
[32] H. Bangui, M. Ge, and B. Buhnova, "Exploring big data clustering algorithms for Internet of Things applications," in *Proc. 3rd Int. Conf. Internet Things, Big Data Secur.*, 2018, pp. 269–276.
[33] M. Thibaud, H. Chi, W. Zhou, and S. Piramuthu, "Internet of Things (IoT) in high-risk environment, health and safety (EHS) industries: A comprehensive review," *Decis. Support Syst.*, vol. 108, pp. 79–95, Apr. 2018.
[34] P. M. Kumar, U. Gandhi, R. Varatharajan, G. Manogaran, R. Jidhesh, and T. Vadivel, "Intelligent face recognition and navigation system using neural learning for smart security in Internet of Things," *Cluster Comput.*, vol. 22, pp. 7733–7744, Nov. 2017.
[35] S. He, S.-H. G. Chan, L. Yu, and N. Liu, "Fusing noisy fingerprints with distance bounds for indoor localization," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, Apr. 2015, pp. 2506–2514.
[36] S. Kumar, M. A. Qadeer, and A. Gupta, "Location based services using Android (LBSOID)," in *Proc. IEEE Int. Conf. Internet Multimedia Services Archit. Appl. (IMSAA)*, Dec. 2009, pp. 1–5.
[37] C. Beder and M. Klepal, "Fingerprinting based localisation revisited: A rigorous approach for comparing RSSI measurements coping with missed access points and differing antenna attenuations," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat. (IPIN)*, Nov. 2012, pp. 1–7.
[38] Y. Shen, S. Mazuelas, and M. Z. Win, "Network navigation: Theory and interpretation," *IEEE J. Sel. Areas Commun.*, vol. 30, no. 9, pp. 1823–1834, Oct. 2012.
[39] A. Fahad, N. Alshatri, Z. Tari, A. Alamri, I. Khalil, A. Y. Zomaya, S. Foufou, and A. Bouras, "A survey of clustering algorithms for big data: Taxonomy and empirical analysis," *IEEE Trans. Emerg. Topics Comput.*, vol. 2, no. 3, pp. 267–279, Sep. 2014.
[40] C. D. Nguyen, D. T. Nguyen, and V. H. Pham, "Parallel two-phase K-means," in *Proc. Int. Conf. Comput. Sci. Appl.* Berlin, Germany: Springer, Jun. 2013, pp. 224–231.
[41] A. Vathy-Fogarassy, A. Kiss, and J. Abonyi, "Improvement of Jarvis–Patrick clustering based on fuzzy similarity," in *Proc. Int. Workshop Fuzzy Log. Appl.* Berlin, Germany: Springer, Jul. 2007, pp. 195–202.
[42] D. Pandove and S. Goel, "A comprehensive study on clustering approaches for big data mining," in *Proc. 2nd Int. Conf. Electron. Commun. Syst. (ICECS)*, Feb. 2015, pp. 1333–1338.
[43] W. Wang, J. Yang, and R. Muntz, "STING: A statistical information grid approach to spatial data mining," in *Proc. VLDB*, vol. 97, 1997, pp. 186–195.
[44] Y. Li, H. Liu, G.-P. Liu, L. Li, P. Moore, and B. Hu, "A grouping method based on grid density and relationship for crowd evacuation simulation," *Phys. A, Stat. Mech. Appl.*, vol. 473, pp. 319–336, May 2017.
[45] C. Lin, Y. Yang, and T. Rutayisire, "A parallel cop-K means clustering algorithm based on MapReduce framework," in *Knowledge Engineering and Management*. Berlin, Germany: Springer, 2011, pp. 93–102.
[46] P. Goyal, S. Kumari, S. Sharma, D. Kumar, V. Kishore, S. Balasubramaniam, and N. Goyal, "A fast, scalable SLINK algorithm for commodity cluster computing exploiting spatial locality," in *Proc. IEEE 18th Int. Conf. High Perform. Comput. Commun.; IEEE 14th Int. Conf. Smart City; IEEE 2nd Int. Conf. Data Sci. Syst. (HPCC/SmartCity/DSS)*, Dec. 2016, pp. 268–275.
[47] M. R. Faheem, T. Anees, and M. Hussain, "Keywords and spatial based indexing for searching the things on web," *KSII Trans. Internet Inf. Syst. (TIIS)*, vol. 16, no. 5, pp. 1489–1515, 2022.
[48] J. Hwang and Y. Yoon, *Data Analytics and Visualization in Quality Analysis Using Tableau*. Boca Raton, FL, USA: CRC Press, 2021.
[49] A. C. Jeba Malar, G. Kousalya, and M. Ma, "Markovian model based indoor location tracking for Internet of Things (IoT) applications," *Cluster Comput.*, vol. 22, no. 5, pp. 11805–11812, 2019, doi: 10.1007/s10586-017-1494-z.
[50] Z. Fang, J. Wang, J. Du, X. Hou, Y. Ren, and Z. Han, "Stochastic optimization-aided energy-efficient information collection in internet of underwater things networks," *IEEE Internet Things J.*, vol. 9, no. 3, pp. 1775–1789, Feb. 2022, doi: 10.1109/JIOT.2021.3088279.
[51] Z. Huang and S. Wang, "Multilink and AUV-assisted energy-efficient underwater emergency communications," *IEEE Internet Things J.*, vol. 10, no. 9, pp. 8068–8082, May 2023, doi: 10.1109/JIOT.2022.3230322.
[52] A. B. Gavali, M. V. Vaze, and S. A. Ubale, "Energy optimization using swarm intelligence for IoT-authorized underwater wireless sensor networks," *Microprocessors Microsyst.*, vol. 93, Sep. 2022, Art. no. 104597,
[53] *GitHub, Xml Document Parser for PHP and Laravel*. Accessed: Sep. 2022. [Online]. Available: https://github.com/orchestral/parser
[54] *GitHub, Json-Ld Processor for PHP*. Accessed: Feb. 2015. [Online]. Available: https://github.com/lanthaler/JSONLD
[55] Accessed: Sep. 2022. [Online]. Available: https://encyclopedia.pub/item/revision/6672a32c89cf2a1eb0b8476ee19752c8
[56] N. Chaudhry, M. M., Yousaf, and M. T. Khan, "Indexing of real time geospatial data by IoT enabled devices: Opportunities, challenges and design considerations," *J. Ambient Intell. Smart Environ.*, vol. 12, no. 4, pp. 281–312, Jan. 2020.
[57] S. Alamri, D. Taniar, and M. Safar, "Indexing moving objects in indoor cellular space," in *Proc. 15th Int. Conf. Netw.-Based Inf. Syst.*, Sep. 2012, pp. 38–44.
[58] J. P. Jeong, S. Yeon, T. Kim, H. Lee, S. M. Kim, and S. C. Kim, "SALA: Smartphone-assisted localization algorithm for positioning indoor IoT devices," *Wireless Netw.*, vol. 24, no. 1, pp. 27–47, 2018.
[59] G. Oguntala, R. Abd-Alhameed, S. Jones, J. Noras, M. Patwary, and J. Rodriguez, "Indoor location identification technologies for real-time IoT-based applications: An inclusive survey," *Comput. Sci. Rev.*, vol. 30, pp. 55–79, Nov. 2018.
[60] P. Y. Taser and V. K. Akram, "Machine learning techniques for IoT-based indoor tracking and localization," in *Emerging Trends in IoT and Integration With Data Science, Cloud Computing, and Big Data Analytics*. Hershey, PA, USA: IGI Global, 2022, pp. 123–145.
[61] J. H. Huh and K. Seo, "An indoor location-based control system using Bluetooth beacons for IoT systems," *Sensors*, vol. 17, no. 12, p. 2917, 2017.
[62] *An Indoor Location-Based Control System Using Bluetooth Beacons for IoT Systems*. Accessed: Dec. 2017. [Online]. Available: https://qualitastech.com/image-processing/differences-between-machine-learning-and-rule-based-systems/
[63] Y. Tang, Z. Pan, W. Pedrycz, F. Ren, and X. Song, "Viewpoint-based kernel fuzzy clustering with weight information granules," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 7, no. 2, pp. 342–356, Apr. 2023, doi: 10.1109/TETCI.2022.3201620.
[64] P. S. Farahsari, A. Farahzadi, J. Rezazadeh, and A. Bagheri, "A survey on indoor positioning systems for IoT-based applications," *IEEE Internet Things J.*, vol. 9, no. 10, pp. 7680–7699, May 2022, doi: 10.1109/JIOT.2022.3149048.
[65] G. Guerrero-Ulloa, A. Andrango-Catota, M. Abad-Alay, M. J. Hornos, and C. Rodríguez-Domínguez, "IdeAir: IoT-based system for indoor air quality control," in *Proc. Int. Symp. Ambient Intell.*, 2023, pp. 197–206.
[66] W. Zhao, A. Goudar, and A. P. Schoellig, "Finding the right place: Sensor placement for UWB time difference of arrival localization in cluttered indoor environments," *IEEE Robot. Autom. Lett.*, vol. 7, no. 3, pp. 6075–6082, Jul. 2022, doi: 10.1109/LRA.2022.3165181.
[67] M. Z. Chaari, R. Al-Rahimi, and A. Aljaberi, "Real-time monitoring of indoor healthcare tracking using the Internet of Things based IBeacon," in *Proc. 18th Int. Conf. Remote Eng. Virtual Instrum.* Cairo, Egypt: The British Univ. in Egypt, Cairo, 2022, pp. 332–342.
[68] M. W. P. Maduranga and R. Abeysekera, "Bluetooth low energy (BLE) and feed forward neural network (FFNN) based indoor positioning for location-based IoT applications," *Int. J. Wireless Microw. Technol.*, vol. 12, no. 2, pp. 33–39, Apr. 2022.
[69] H. Wang, C. C. Tan, and Q. Li, "Snoogle: A search engine for pervasive environments," *IEEE Trans. Parallel Distrib. Syst.*, vol. 21, no. 8, pp. 1188–1202, Aug. 2010.

[70] K.-K. Yap, V. Srinivasan, and M. Motani, ''MAX: Human-centric search of the physical world,'' in *Proc. 3rd Int. Conf. Embedded Networked Sensor Syst.*, Nov. 2005, pp. 166–179.

[71] Z. Ding, Z. Chen, and Q. Yang, ''IoT-SVKSearch: A real-time multimodal search engine mechanism for the Internet of Things,'' *Int. J. Commun. Syst.*, vol. 27, no. 6, pp. 871–897, Jun. 2014.

**MUHAMMAD REHAN FAHEEM** was born in Pakistan. He received the B.Sc. degree in information technology from the Islamia University of Bahawalpur, Pakistan, in 2012, and the M.Phil. degree in computer science from NCBAE, Lahore, Pakistan, in 2016. He is currently pursuing the Ph.D. degree with the University of Management and Technology, Lahore, under the supervision of Dr. Muzammil Hussain and the Co-Supervision of Dr. Tayyaba Anees. He has strong analytical and problem solving abilities. His research interests include the Internet of Things, the Web of Things, image processing, information retrieval, computer graphics, computer vision, graphics modeling, and machine learning.



**TAYYABA ANEES** was born in Pakistan. She received the Ph.D. degree from the Vienna University of Technology, Vienna, Austria, in 2012. Her Ph.D. dissertation is in the area of service-oriented architecture and web services availability domain. She was the Project Assistant with the Vienna University of Technology for four years. She is currently the Program Head Software Engineering/an Assistant Professor with the Software Engineering Department, School of Systems and Technology, University of Management and Technology, Lahore. Her research interests include service-oriented architecture, web services, the Web of Things (WOT), semantic web, software availability, software safety, software fault tolerance, and real-time data warehousing.

**MUZAMMIL HUSSAIN** received the B.Sc. degree in computer science from COMSATS University Islamabad, in 2013, and the Ph.D. degree from the University of Malaya, Kuala Lumpur, Malaysia. From 2013 to 2017, he was a Research Associate with the Department of Computer Systems and Technology, Faculty of Computer Science and Information Technology, University of Malaya. He is currently an Assistant Professor with the Department of Computer Science, School of Systems and Technology, University of Management and Technology, Lahore, Pakistan. He is also the Director of graduate studies and research with the School of Systems and Technology, University of Management and Technology. He led or a member of many funded research projects and he has published more than 46 research papers in prestigious international conferences and journals. His research interests include web services, the Web of Things, network security, fintech security, blockchain, android security, bioinformatics, the IoT security, SDN security, and internet security. He has been ranked in top 2% scientists of the world by Stanford University rankings, in 2022. He was awarded with Gold Medal, for his B.Sc. degree, and the Bright Sparks Program Scholarship from the University of Malaya, for his Ph.D. studies.

**ALLAH DITTA** received the M.Sc. degree in computer science and technology from Quaid-i-Azam University (Q.A.U), Islamabad, Pakistan, in 2012, and the Ph.D. degree in computer science and technology from the College of Computer Science, Beijing University of Technology, China, in 2017. In 2017, he joined as an Assistant Professor with the Department of Information Sciences, Division of Science and Technology, University of Education, Lahore, Pakistan. He has authored several articles in well-recognized journals, such as *Journal of King Saud University, Computer and Information Sciences*, IEEE Access, *Sensors*, *Computers, Materials and Continua*, and *Wireless Communications and Mobile Computing*. His research interests include machine learning, information security, digital steganography, cryptography, network security, and wireless sensor networks.

**HANI ALQUHAYZ** received the bachelor's degree in computer science and the master's degree in information systems management from King Saud University, and the Ph.D. degree in computer science from De Montfort University, U.K. He is currently an Associate Professor with the Computer Science Department, College of Science, Majmaah University, Saudi Arabia. He has authored several articles in high-impacted journals, such as IEEE Access, *Sensors*, and *Wireless Communications and Mobile Computing*. His research interests include wireless security, scheduling, image processing, the IoT, security and privacy, data mining, machine learning, and artificial intelligence.

**MUHAMMAD ADNAN KHAN** received the B.S. and M.S. degrees from International Islamic University, Islamabad, Pakistan, and the Ph.D. degree from ISRA University, Islamabad. He is currently an Associate Professor with the School of Informational Technology, Skyline University College, Sharjah, United Arab Emirates, and the Riphah School of Computing and Innovation, Faculty of Computing, Riphah International University, Lahore Campus, Lahore, Pakistan. He is also an Assistant Professor with the Department of Software, Faculty of Artificial Intelligence and Software, Gachon University, Republic of Korea. He has worked in various academic and industrial roles in Pakistan and Republic of Korea. He has been teaching graduate and undergraduate students in computer science and engineering for the past 14.5 years. He is guiding six M.Phil. scholars and five Ph.D. scholars. He has published more than 230 research articles with cumulative JCR-IF more than 585 in reputed international journals and international conferences. His research interests include machine learning, MUD, image processing and medical diagnosis, and channel estimation in multi-carrier communication systems using soft computing. For his Ph.D. degree, he received the Scholarship Award from the Higher Education Commission, Islamabad, in 2016, and the Scholarship Award from the Punjab Information and Technology Board, Government of Punjab, Pakistan, for his B.S. and M.S. degrees.