IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received July 2, 2019, accepted July 19, 2019, date of publication July 29, 2019, date of current version August 14, 2019.

*Digital Object Identifier 10.1109/ACCESS.2019.2931659*

# Search Engine for the Internet of Things: Lessons From Web Search, Vision, and Opportunities

**FAN LIANG, CHENG QIAN, WILLIAM GRANT HATCHER, AND WEI YU**
Department of Computer and Information Sciences, Towson University, Towson, MD 21252, USA

Corresponding author: Wei Yu (wyu@towson.edu)

This work was supported in part by the U.S. National Science Foundation (NSF) under Grant CNS 1350145 and in part by the University System of Maryland through the Wilson H. Elkins Professorship Award Fund. Any opinions, findings and conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funding agency.

**ABSTRACT** With the development of the Internet of Things (IoT), massive numbers of IoT devices (smart sensors, cameras, phones, and so on) have been deployed and utilized in various environments, supporting numerous smart-world applications. Those devices communicate with computing infrastructure servers over network infrastructures to send relevant collected data in order to link physical objects to the cyber world. As the number of IoT devices increases rapidly, the volume of collected data likewise increases prodigiously. Thus, how to search for and through specific IoT datasets among the enormous amount of data become critical issues for potential data consumers. Moreover, various IoT devices and applications establish IoT-based systems, also known as the smart-world systems or smart cyber–physical systems (CPS), such as smart grids, smart transportation, smart healthcare, smart cities, smart homes, and smart manufacturing systems, among others. However, the individual CPS are independently designed and deployed, such that they collect and analyze data independently, with no information sharing or interconnection, raising serious challenges in searching for valuable information. Thus, in order to efficiently and precisely utilize the IoT datasets, suitable search techniques designed for the IoT environments are fundamental. In this paper, we first summarize popular web search techniques and survey existing research on the search and analysis related to the IoT. We then outline the opportunities and challenges of the IoT search techniques. Furthermore, we propose a problem space for the IoT search techniques and provide a clear view of potential future research directions.

**INDEX TERMS** Internet of Things, search engine, system architecture, challenges and research opportunities, data mining and analytics.

## I. INTRODUCTION

The Internet of Things (IoT) is currently a highly active topic, with related techniques and applications being developing at extraordinary speed, in a variety of domains such as city infrastructure, healthcare, energy, industry, and others [5]–[11]. As of April 03, 2019 [12], there are over 132 million Internet-connected IoT devices deployed and in use worldwide. IoT enables the connectivity of physical things and cyber systems, and the massive number of IoT devices already in the wild are collecting and analyzing valuable information concerning society, industry, and the environment [13], [14] without human intervention. Moreover, Cyber-Physical Systems (CPS), based on a vertical architecture (i.e., application, networking, and physical), leverage IoT and the generated datasets, and creates a closed loop process, in which data is collected, analyzed, and decision-making is resolved upon [6]. CPS are able to monitor and operate by analyzing IoT data to make systems smart and partially or fully autonomous. Therefore, the key value of IoT and CPS is reflected in the datasets that they collect and analyze.

Because of the increase in deployment of smart devices, the amount of collected data is constantly increasing, benefiting the data analysis process and yielding more comprehensive results. However, this induces obvious pressure in the targeted and rapid search for specific data and datasets from the massive volume collected. Specifically, IoT datasets are quite massive, but not all data is valuable for providing guidance in decision-making. Furthermore, specific analysis processes require different datasets or dataset combinations for different purposes and applications. Thus, a search service

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

**TABLE 1.** Properties of existing IoT search engines.

| Name | Data Source | Purpose | Function |
|---|---|---|---|
| ***Shodan*** [1] | IoT devices via crawlers | Assist cyber-security researchers to analyze the influence of vulnerabilities online | Vulnerability search, IoT device meta-data search |
| ***Thingful*** [2] | IoT devices via crawlers and open source data from third-party organizations | Assist users to retrieve information on weather, parking lots, traffic, etc. throughout the world. | Meta-search to different organizations, IoT device meta-data search |
| ***Censys*** [3] | IoT devices via crawlers | Assist cyber-security researchers to analyze the influence of vulnerabilities online | Vulnerability search, IoT device meta-data search |
| ***Reposify*** [4] | IoT devices via crawlers | Assist cyber-security researchers to detect vulnerabilities in organizations | Vulnerability search, IoT device meta-data search |

for IoT data is essential. Although there are some IoT applications with preliminary search functions, these have numerous limitations, such as only being applicable for searching specific datasets defined by the applications, and only being suitable for searching static datasets. Therefore, a comprehensive IoT search engine becomes necessary to improve the performance and value of IoT and CPS more broadly [15].

Furthermore, IoT data is abundant and diverse, since IoT devices are deployed in different CPS and collect data constantly. The volume, velocity, and variety of IoT data make the datasets more valuable as well [16]. Thus, utilizing IoT data efficiently and comprehensively is a particular goal of the IoT field of research. However, collection and reassembly of the required datasets is a challenging issue because, using only one fixed or closed application, it is difficult or impossible to access different types of IoT datasets, which are collected by different CPS. For instance, the autopilot applications of smart vehicles not only require traffic information, but also parking lot, gas station/charging station, and weather information, among others. Nonetheless, this information is provided by many different CPS and there is no unified interface designed in the application layer of the IoT infrastructure to allow data consumers to access various IoT datasets from different CPS. The CPS only provide data and information services to the consumers by leveraging their own applications and infrastructures. Those distinct and separate applications raise a critical issue of unified management, access, and leveraging of the IoT datasets.

In terms of development, the IoT search engine is in the initial stages, at present. Indeed, while there are several popular IoT search engines that provide search services, including Shodhan [1], Thingful [2], Censys [3], and Reposify [4], similar to the initial stages of web search engines, all of them have some limitations. For instance, they are only able to search online IoT devices with open-application programming interfaces (APIs), as well as some static data sources which have been uploaded to the search engine's accessible servers. These existing search engines, as shown in Table 1 below, obviously do not satisfy the requirements of IoT or a fully-realized IoT search engine. Additionally, a number of studies have focused on IoT search techniques. For instance, a general view of search techniques was first proposed by Romer *et al.* [17], and further work by Zhang *et al.* [18] compared IoT search techniques with traditional web search techniques and summarized the differences and challenges. However, the study of and recommendations for how to integrate existing CPS and leverage IoT datasets efficiently remain unclear.

In this paper, we briefly summarize existing web search techniques, and then survey related studies on IoT search techniques in detail. Based on this systematic study, we then propose a problem space for IoT search techniques and provide a clear view of potential research directions. The major contributions of this paper are listed as follows:

- **Problem Space of IoT Search Engines:** Traditional web search engines are mature, and have some similarity to IoT search engines from a functional point of view. Thus, reviewing the development roadmap of web search engines is helpful to understand the development of IoT search engines. Therefore, in this study, we first review the development of the Google search engine and deeply study existing IoT search techniques. Then, we discuss the opportunities and challenges of the IoT search engine from the perspectives of the IoT framework, connections, protocols, and applications. Finally, based on the study, we propose a problem space for IoT search engines.
- **Further Research Directions:** Reflecting on the development roadmap of the Google web search engine, we propose possible future research directions in IoT search in the contexts of distributed computing, machine learning, interaction between CPS, resource integration, and security and privacy. Specifically, since IoT search engines are comprehensive systems, they require support from multiple sub-systems to realize and improve their performance. Therefore, our work not only focuses on IoT search engines themselves, but also considers related supporting techniques, which are able to improve and promote the development of IoT search engines, such as edge computing and machine learning. Combined with the necessary related techniques, we provide a clear picture of the IoT search engine broadly.

The remainder of this paper is organized as follows: In Section II, we briefly introduce web search engines, including an overview of the historical demands and context of web search engines, generalizing their characteristics, as well as introducing web search architectures and workflow. In Section III, we discuss the motivations for introducing search engines into IoT systems. Based on the discussion, we systematically study existing IoT search techniques and provide a comprehensive survey on the fundamentals of IoT search and relevant techniques. In Section IV, we first expound the relationships between CPS and IoT, and then propose our view, namely that CPS have needs that require IoT search services and that CPS automation is one of the important motivations for developing IoT search engine. Then, we present the problem space of IoT search engines. According to the problem space, we summarize the opportunities and challenges of IoT search. In Section V, we introduce possible future directions for IoT search engines in detail. Finally, we summarize or efforts and provide some final comments in Section VI.

## II. OVERVIEW OF WEB SEARCH ENGINE

In this section, we first review the development roadmap of the web search engine as a representative example, from which we can consider similarities with current and future IoT search engine development. To do so, we deeply investigate the most widespread, recognizable, and successful web search engine: Google (92.04% of market share worldwide [19]), and extract milestones and key search techniques as the references for the IoT search engine.

### A. DEMANDS AND HISTORY

The Internet has developed at an extremely fast pace, and has changed significantly since the 1990s. Information dissemination has almost entirely changed from physical to digital media. Thus, massive information is stored in digital format (documents, multimedia, database, etc.) and covers all fields of industry, society, and the environment. Similar to the directory for a book or a library, a search tool is necessary to assist users in finding the information that they are looking for, which is the basic demand that promoted the development of the web search engine. The development of the web search engine progressed primarily in four key stages as considered from a functional point of view. These are website navigation, text retrieval, integration and analysis, and intelligence and customization, which we detail in the following.

#### 1) WEBSITE NAVIGATION

In its initial stage, the search engine functioned as website that hosted records of the addresses of the most popular websites, and was developed for guiding users to find the information they needed. This kind of website is called a website navigator and it is the prototype of web search engines. The website navigator satisfies the requirements of users who browse and seek specific information from the Internet. The users only need to remember the address of the website navigator, and from the website navigator, the users are able to access the specific websites by clicking the hosted hyperlinks, which are recorded in the website navigator. By utilizing the website navigator, the users are not required to remember all the necessary website addresses, a method that is convenient and fast.

#### 2) TEXT RETRIEVAL

At the next stage of complexity, users are able to send queries to the search engine and obtain the relevant information. The search engine, according to the keywords in the queries, searches File Transfer Protocol (FTP) servers and provides feedback to the users. In this stage, the information retrieval model of the search engine includes the boolean model, probability model, and the vector space model, while the search categories are limited to text and documents. Examples include Alta Vista, Excite, and Archie [20].

#### 3) INTEGRATION AND ANALYSIS

In this stage, the search engine analyzes the popularity of the website, and when users send queries to the search engine, the returned results are ordered by popularity. Furthermore, instead of only returning some website links, the search engine integrates the results into one interface with descriptions, images, and highlighted content relevant to the query input. This approach highly improves the user experience. Google was one of the first web search engines to integrate this search model. Moreover, based on this model, Google proposed and developed link analysis algorithms to improve search performance [21]. Today, link analysis algorithms have been widely used in web search engines.

#### 4) INTELLIGENCE AND CUSTOMIZATION

Current web search engines can be categorized into this stage of development. Here, search engines return queries organized not only according to keywords, but also by leveraging big data analysis and user data for a better understanding of query language, usage, and context [22]. For instance, when the search engine receives a query with the keyword ''apple'', the search engine first analyzes the browsing history of the user to determine whether the user is a potential iPhone consumer or a fruit supplier. Then, the search engine returns customized information for different users. Thus, in this stage, the search engines focus on how to leverage user data to fully understand the demands of the particular user. In more detail, utilizing data analysis mechanisms, such as machine learning and deep learning, the search engines are able to gather relevant information to undertake complex searches [23]. This complexity is amplified by user data generated via mobile devices, through which search engines are able to gradually extract characteristics and identifying features of the user. This is also the embryonic stage for IoT search engines today.

### B. CHARACTERISTICS

In this section, we discuss the characteristics of web search engines, which are fundamental to the design of future

IoT search engines. We generalize three basic characteristics of a web search engine, as follows.

#### 1) RELIABILITY

Reliability is the most important characteristic of a real-world search engine. For instance, the Google search engine processes over 75 thousand searches and 72 thousand GB of content in a single second [24]. Thus, failures of servers or malfunctions can have an impact that is near unfathomable.

#### 2) CRAWLING

Crawling is a fundamental approach for search engines to extract information from multiple websites. As introduced by Manning *et al.* [25] in their basic crawling algorithm, crawlers first initialize a queue of uniform resource locators (URLs) and then verify whether the web page can be crawled. Utilizing the crawling algorithm, the web search engines are able to access websites in the World Wide Web (Internet), and, after crawling, the search engines associate the URL with a unique document ID [26]. When the search engine receives a query, it is then able to match the document ID and keyword in a very short period of time. Moreover, Brin and Page [26] introduced the forward and invert index. The forward index is a mechanism to map documents to keywords, first retrieving keywords from the document and then identifying the keyword that matches the query. This is obviously not efficient for the searching process. Thus, the invert indexing, as the key enabling component, associates the keyword with all the documents which contain it. In this way, users can quickly obtain accurate searching results.

#### 3) RANKING

Query return and page ranking are approaches to improve user experience. Ranking algorithms that function on user feedback and keyword occurrence rate can assist search engines to optimize the sequence of results. For example, the Google search engine utilizes the content score, popularity score, and overall score, three key parameters to optimize the ranking of results [27].

### C. ARCHITECTURES AND WORKFLOW

In this section, we leverage the Google search engine as an example. We first introduce the key platform and architectures of the Google search engine, and then present the workflow starting with new information detection and ending at response to queries.

#### 1) ARCHITECTURES

Google is the most popular web search engine to date, with 92% of web queries utilizing Google search [28]. Thus, the architectures of the Google search engine are able to represent the development tendencies of web search techniques well, especially as the engine has had to evolve to support this large user base. The Google search system consists of several major components, including Google File System (GFS), MapReduce, and BigTable. These components are responsible for responding to queries and optimizing system performance. In the following, we will discuss these components in detail.

- **Google File System:** This powerful database system is required for general data center retrieval. Particularly, the Google search engine finds itself in a critical situation, as it must cope with more than ten billion web documents from around the world. Indeed, if it were to utilize a traditional database system, it would be unusable, given the latency necessary to search for specific documents. Thus, the Google File System (GFS) was designed to deal with a large amount of information. When the crawlers gather new web data, Google re-organizes the information into a file and compresses the file into chunk blocks to reduce the size (64 MB per chunk) [29]. GFS is designed around retrieval technology that includes GFS servers and Chunk servers. The GFS is a distributed system and the master node is responsible for maintaining the name space of the system, access control information, mapping files to blocks, and the current location of the blocks. The master node communicates with several chunk servers by utilizing heartbeat signals, and gathers the status of the chunk servers. The chunk servers store data blocks and copy the data blocks to three different servers for reliability and redundancy.
- **MapReduce:** Google's MapReduce focuses on how to optimize retrieval speed and obtain specified data from massive volumes of information. To be specific, the query is first duplicated to multiple copies and sent to idle workers for processing. Meanwhile, the map and reduce tasks are distributed to the working clusters. The map tasks transform the queries into a key/value type for input to the map function, with the output then stored into memory. The reduce tasks act according to the key/value output to find the data and return a result file [30]. The result file is the initial search result awaiting further optimization.
- **BigTable:** The BigTable architecture is a storage system that is built on top of the GFS, which not only takes responsibility for storing the structured data, but also optimizes data management and load balancing decisions. BigTable technology is widely utilized in different Google products, such as Google search and Google Maps. The BigTable database is usually divided into small pieces, which are named tablets. The tablets are deployed on the different computing devices in the GFS cluster in order to store the massive amount of data. Generally speaking, the BigTable architecture includes a master server that takes responsibility for assigning tablets to tablet servers and balancing tablet-server load. In addition, tablet servers execute the read and write operations to the tablets and monitor the loads for each tablet. Then, the tablet servers split the incoming data and write the data to idle tablets in order to balance

the loads. Furthermore, the tablets are working independently, and do not require constant communication with the master node. Thus, the BigTable approach can reduce the communication loads of master servers [31], [32].

#### 2) WORKFLOW

We now brief the workflow of conventional web search engines.

- **Before the Query:** Web search engines utilize a crawler tool to browse and traverse the Internet. The main task of the crawler is detecting updated information, collecting the new information, and storing the new information to storage (denoted the ''collection pool''). In the second step, search engines create the index for the collected information and prepare for searching. Taking the Google search engine as an example, we further discuss the details of the ''Before the Query'' phase. When users update data in their website, blog, etc., the Google crawler detects that the information is updated. Then, judgment mechanisms for Google's crawler determine whether to collect the information or not. For instance, the crawler will not crawl the information if no URL link to the website exists or there is no permission to access the website (defined in the strategy documents). Meanwhile, the Google crawler utilizes URLs as road signs to travel the Internet. Thus, Google has also developed a mechanism to analyze and identify the value of the URLs. This mechanism labels low-value URLs as ''nofollow'', and the crawler does not access the website. After new information is collected by the crawler, two steps remain. First, titles and link data are created for the webpages using a breadth-first search. Then, the content of the named pages are stored with an index table, which is used to perform long tail, personalized, and depth-first searches with low frequency.
- **After the Query:** There are generally a few steps from when the users send the quires to when the search engine returns refined results (note that preliminary results are not presented directly to the user) [27]: (i) Users send queries to the search engine, which puts the queries into multiple parallel control processes to be sent to different search engine components to request service. (ii) Google provides some suggestions based on the keywords of the queries. For instance, based on a user's searching history, a keyword of ''apple'' could be related to iPhone and other Apple digital products or to fruit suppliers and purchasers. (iii) Preliminary results that match the keywords of the queries are collected. (iv) Refined query results, which, in the case of Google, is less than 1000 results, will be localized based on the determined location of the user and will be arranged geographically. Furthermore, Google optimizes the searching results based on ranking, location, personality, and tendency before showing the results to the user on their website.

## III. IOT SEARCH ENGINE

In this section, we first illustrate the motivations of the design of IoT search engines. Then, we systematically study existing IoT search engine techniques from the perspectives of components and architectures. Finally, we classify IoT search engines from several perspectives.

### A. DESIGN MOTIVATIONS OF IOT SEARCH ENGINE

We now consider the motivations of developing and implementing IoT search capabilities. These motivations include data sharing, resource integration, and artificial intelligence.

#### 1) DATA SHARING

Due to the development of IoT and the increasing volume of IoT devices, the amount of IoT data is increasing at an unprecedented rate. Furthermore, this data is updated continuously, and can dynamically describe the status of the related systems in order to provide a more precise and accurate description of a system's state than stale static data. Thus, the three V's (volume, velocity, and variety) of big data are realized in IoT, demonstrating its the potential value [33]. However, searching for specific IoT data is a challenging problem for data consumers, which slows down the data sharing process and decreases data value. Therefore, similar to web search engines, there is an urgent demand to have a searching engine for IoT to provide the query resolution services to assist users in finding relevant IoT data efficiently.

#### 2) RESOURCE INTEGRATION

In general, IoT data is stored in different CPS independently, which cannot communicate with each other. For instance, in a smart grid system, a smart meter uploads energy consumption data to the operation servers in the grid, supporting a number of services, including demand response, dynamic pricing, and integration of renewable energy resources, among others [34], [35]. Meanwhile, a weather monitoring system collects temperature and humidity data from sensors and uploads the data to weather monitoring servers. In reality, those two systems and their data are dependent, since the energy consumption data is impacted by the weather conditions. Electric vehicles are another resource integration example that belongs to the key components in both the smart grid and smart transportation system [36], [37].

Likewise, the smart city has been developing rapidly to enable the optimization of resources such as waste, traffic, parking, and others, as well as toward improving quality of life [7], [38], [39]. For example, governments utilize IoT devices to monitor traffic conditions and violations in major cities [40], [41]. When pedestrians cross the street illegally, high-resolution cameras are activated and capture an image of the pedestrian violating the law. The portrait is sent to different databases, such as citizen information databases, public security databases, police databases, etc., in order to determine the identity of the pedestrian. During the process, the system not only needs the information from the

cameras, but it also needs the reference information to help in the identification process, such as traffic light information, location information, and others. Based on these examples, we know that real production systems typically send multiple queries to different CPS to obtain the necessary datasets, increasing the difficulty of utilizing and analyzing the data comprehensively. Thus, it is necessary to design a unified interface or platform to manage queries for multiple CPS and support the inter-operation of systems.

#### 3) ARTIFICIAL INTELLIGENCE

Currently, utilizing and analyzing IoT data accelerates the development of Artificial Intelligence (AI). These increasingly popular and powerful data analysis tools, such as Machine Learning and Deep Learning, also increase the access demands on IoT data [42]–[46]. Unlike the web search engine, in IoT systems, more data exchanges occur between smart devices than between users and devices. Taking the smart grid as an example, a large number of smart meters and sensors are deployed in the smart grid to monitor the status of power generation and distribution [47], [48]. In some cases, the smart meters collect abnormal data, and in order to identify anomalies, some other information or reference data is required. To this end, the smart grid system automatically sends queries to search engines to request weather, public safety, and electricity use information that are related to the location of smart meters. Then, depending on the comprehensive analysis, the smart grid system automatically adjusts the power supply to adapt to the assessed situation.

As another example, smart transportation is of increasing interest due to its potential to reduce congestion and improve traveler safety [49]–[51]. In smart transportation, autopilot systems in smart vehicles utilize sensors and data links to observe their surroundings and communicate with other facilities. Each autopilot vehicle sends queries to the search engines in real-time to obtain traffic information, parking information, weather information, etc., to optimize the route. In addition, the autopilot vehicles are able to measure the volume of fuel (gas or electricity) within themselves to assess the need for additional fuel and send queries to obtain location information of gas or charging stations. As these types of events occur quite frequently in IoT, an IoT search engine that can support many such transactions is required to provide these services.

### B. FUNDAMENTALS OF IOT SEARCH ENGINE

We now consider the fundamentals of IoT search engines, which we subdivide into components and architectures.

#### 1) COMPONENTS OF IOT SEARCH ENGINES

An IoT search engine is designed for searching a set of IoT resources, including IoT data and devices. Similar to web search engines, IoT search engines respond to queries, returning the IoT resources, IoT data, or combinations of both, as necessary. Here, we introduce the fundamental components of the IoT Search engine.

- **IoT Resources:** In IoT, physical objects embedded with computing and networking components become smart resources (smart sensors, actuators, storage, and networking devices, among others) with a digital form. All IoT devices are abstracted as IoT resources in the IoT search process. Moreover, the IoT search engine generally utilizes two methods to find IoT resources, which are referred to as individual search and cooperative search [52]. To be specific, individual search discovers IoT resources based on the resource ID and the related data type that is generated by the IoT resources. For example, in the smart grid system, electricity suppliers need to know the amount of power usage in a time period at a specific location. To obtain this information, the operational center sends queries to IoT search engine to obtain the power usage data from the smart meters at the target locations. Likewise, public safety departments need to obtain meteorological information, and thus send queries to IoT search engine to obtain temperature and humidity data from the necessary sensors. Meanwhile, cooperative search grabs IoT resource information not only from the resources themselves, but also from third-party information providers. For example, in the smart transportation system, when autopilot vehicles travel along a road, the vehicles send queries to different CPS to obtain comprehensive information, enabling vehicle guidance to the target destinations. Comprehensive information includes traffic information, parking information, gas station locations, and others.
- **IoT Data:** The IoT data is the key search target and is the core component of the IoT search engine. Based on the source of the data, we can classify IoT data into two categories. The first is the data collected by IoT devices, which is the feedback or measurements of the physical world, such as electricity consumption, temperature, humidity, etc. The second is context data, which describes the state and running condition of the IoT devices, such as availability, network latency, battery life, storage space, etc. Specifically, collected data is generally utilized to create a digital model that represents the status or history of related physical environments. Working with digital models, we are able to simulate the physical environments, in order to predict the future tendencies and optimize system performance. In contrast, context data typically represents the running status of devices. This type of data is critical for analyzing system robustness and providing the guidelines for system update. The IoT systems can do self-check operation based on context data from themselves which represents their working status without comparing current data with history data collected by themselves.
- **Search Space:** Similar to the web search engine, a search space is also necessary for IoT search. In IoT search, the search space is a group of IoT resources which have well-defined data structures, and search

algorithms identify the matching resources according to the queries [53]. The search space for IoT search is far greater than traditional web search, since IoT resources are dynamic, updating much faster than web information. In addition, the IoT devices have a deeper hierarchical structure than websites, and thus, the IoT search engines must crawl a greater space than the web search engines. Moreover, unlike a web search engine, an IoT search engine can get results based on users' locations, purposes, and privileges, and has the ability to send the most accurate and reasonable result back to users. In order to increase the searching speed of IoT search engines, related resources can be associated with relationships and characteristics in order to reduce the size of the search space.

- **IoT Query:** Queries in IoT search can be submitted by a human or an IoT device (i.e., machine) itself. The latter is the key difference between the web search query and the IoT search query. In particular, queries must be sent by IoT devices to achieve automation and intelligence (i.e., machine-to-machine communications). In CPS, the systems automatically monitor and control themselves without human interruption or intervention, based on comprehensive information. In this case, to obtain the required information, the IoT devices need to have the ability to send queries to the search engines and obtain feedback. In addition, some CPS are time sensitive. For instance, autopilot systems on airplanes have anti-collision systems that must work fast, accurately, automatically, to monitor the surroundings of the airplane. IoT devices from multiple planes need to communicate and exchange data with each other, including their height, speed, location, and direction, among others. In cases where communication and decisions must occur in real time, the auto-pilot system can take action in a shorter time than pilots, who need to communicate with the control tower before taking action.
- **Edge Computing Nodes:** Distributed computing can offload computation tasks from cloud infrastructures to edge and fog nodes [54]–[56] that are close to the IoT devices. The edge computing nodes are able to execute the search tasks and are easy to extend. In the IoT search system, the edge nodes are utilized to crawl for IoT information and update the index of the search space.
- **Middleware:** The IoT middleware [57] is another important component in the IoT search engine, which acts as the interface between applications and IoT resources. In particular, it hides the heterogeneous IoT resources and provides a simple operating platform for the queries. Additionally, the middleware can be located anywhere there is enough computation power to handle the necessary computations, including IoT gateways, cloud servers, and edge computing nodes, since it is a service provider for IoT. The middleware can also offer common services for applications and application developers by integrating heterogeneous computing and communications devices. Moreover, middleware supports interoperability to enable diverse applications and services to work together and provides an API to developers for interacting and extending the applications.

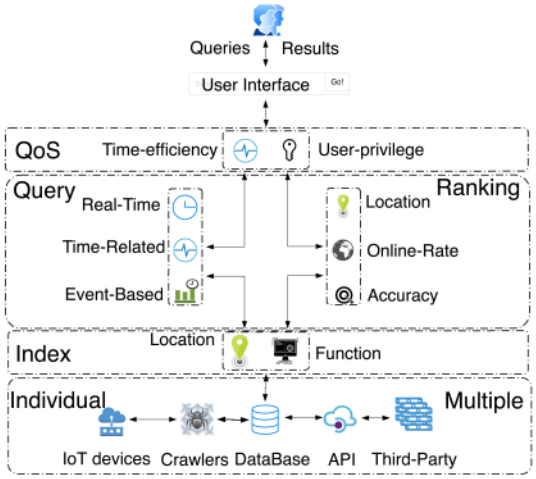


**FIGURE 1.** IoT search engine architecture.

### 2) ARCHITECTURES

Fig. 1 illustrates the basic architectural structures of the IoT search engine, which includes Quality of Service (QoS) module, user interface, query module, ranking module, indexing module, crawlers, database, API, and third-party organization. The QoS module mainly focuses on reducing search times to optimize user experience. It classifies the users into different groups based on user analysis results (behavior analysis, search history analysis, user information analysis, etc.). Based on this analysis, it can reduce the search space for specific users in order to increase the searching speed, and determines which types of information should be returned (dynamic or static). Then, based on the output from the QoS module, the query model executes the search process, which could be a real-time search, time-related search, or event-based search. The crawling results are stored in the database, and the indexing module categorizes the data and maps the data with a unique index in order to increase searching speed. The ranking module retrieves the indexes and organizes the results based on the ranking algorithms.

## C. IoT SEARCH TECHNIQUES

In this subsection, we systematically review existing IoT search techniques and summarize the contributions of relevant research works.

### 1) LOCATION-BASED SEARCH

In IoT systems, the location information of IoT devices is important, as users typically send queries associated with location information. This location information can be presented as geographical coordinates or can be a logical location, such as the distance from another device.


**FIGURE 2.** Real-world map of web cameras on shodan.

One example of such a system was introduced by Liang and Huang [58], who developed the Geospatial Cyber-infrastructure for Environmental Sensing (GeoCENS) architecture in order to stimulate the full potential of the IoT sensors. The GeoCNES was developed based on the Peer-to-peer (P2P) architecture and is a location-aware system which utilizes Sensor Web Long Tail to record and collect the location information of the sensors.

Based on GeoCENS, Mayer *et al.* [59] optimized the infrastructure's lookup mechanism, which leverages location information to improve scalability and load balancing of the IoT search engine. The mechanism has large cache memory and can reduce the response time. In addition, Wang *et al.* [60] developed a geographical location-based sensor discovery architecture. The discovery system utilizes a distributed architecture and associates using geospatial indexing to reduce the search space. Specifically, it uses the *R* tree to index the location information of a given rectangle area and searches for IoT sensors which have the same location information. However, the computation complexity is high. Likewise, Fathy *et al.* [61] proposed an unsupervised machine learning algorithm to optimize the efficiency of sensor discovery. The learning algorithm divides the sensors by geographic information into clusters. In addition, the learning algorithm continuously discovers hidden sensors by measuring the distances between hidden and known sensors.

The Shodan IoT search engine is another location-based search engine, and is popular for finding specific types of IoT devices, such as webcams and routers. Searches on Shodan return hardware information, device status, and device location to the user. With advances in IoT technology, IoT systems have strong demand for centralized management in order to share information and interact with each other. In fact, Shodan preliminarily integrates Supervisory Control and Data Acquisition (SCADA) systems, which can dynamically grab real-time data from various SCADA systems to readily resolve queries [62].

As an example, we use Shodan to search for all the web cameras throughout the world within Shodan's database. Fig. 2 shows the location distributions of web cameras on a world map with additional information regarding country, protocols, and services. As shown in the figure, through Shodan, we can determine the countries that have the most web cameras, the protocols that the web cameras use, and organizations that they belong to. Fig. 3 illustrates the meta-data of some web cameras, including the device information (e.g., geo-location, description, etc.) [63]. Here, let us first look at the web camera with the IP address of 35.178.106.77. From the meta-data on the right, line 4 shows *WWW-Authenticate: Basic realm="Mini Dome IP Camera"*, where "WWW-Authenticate" is a part of HTTP header that defines the authentication method used to get access to a resource [64], "Basic" is a basic authentication type, and "realm" displays a device's hostname. Shodan reveals that this device is a camera by looking into its hostname. In addition, for the camera with IP addresses 73.169.181.32 and 186.217.200.31, we know that both are web cameras through server names (i.e., Server: ip-camera, 220 Network-Camera FTP server).

### 2) CONTENT-BASED SEARCH

This type of search is conducted based on the data content that is collected by a specific target sensor or sensors. First, the IoT search engines analyze the content and map the corresponding index. Then, when queried, the search engines pair the query and content by using the corresponding index, and return the sensor information. Utilizing content-based search, users can search both real-time and historical data.

One example of content-based IoT search was introduced by Truong and Römer [65], who proposed a content-based

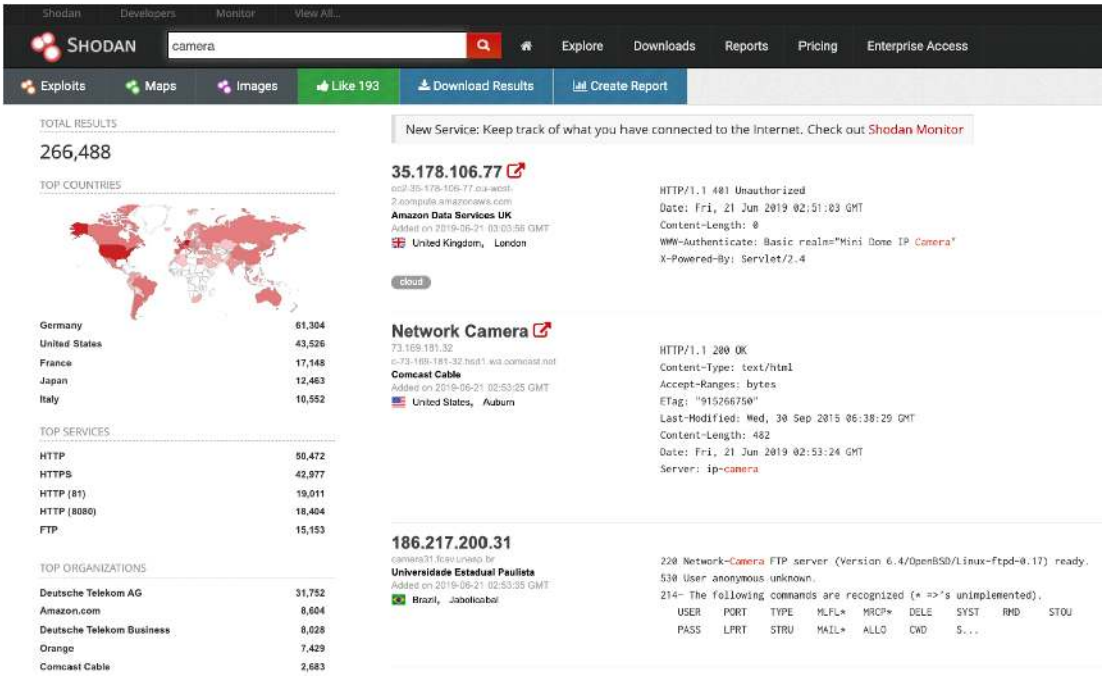


**FIGURE 3.** Meta-data of web cameras on shodan.

sensor search that utilized fuzzy sets to achieve a scalable system. In detail, the proposed search scheme first selects a group of sensors, in which data content is in the range of the query so that the search space can be reduced. Then, only a small group is searched and the result is returned. Likewise, Lunardi *et al.* [66] proposed the content-based search engine COBASEN. COBASEN is based upon the semantic characteristics of the devices, which are abstracted from the data content and assist the users in communicating with the target sensors. Additionally, Elahi *et al.* [67] proposed a prediction model that utilizes a sensor ranking model to arrange the results based on the content. The ranking model first polls the sensors to determine the health status of the sensors based on the content and then arranges the sensors returned to the users. Also, Zhang *et al.* [68] developed a high-efficiency content-based sensor search system which can be considered a prototype of the content-based search. In particular, a sensor state prediction method was designed to estimate the sensor data which can classify sensors into candidate groups in order to increase the search speed. Further, because of the huge communication overhead incurred by IoT search engines accessing all available objects, Zhang *et al.* [69] proposed a low-overhead and high-precision prediction model (LHPM) to improve the sensor search efficiency. First, an approximation scheme was designed to reduce the search energy cost, and a multi-step scheme was developed to accurately estimate the sensor state. Then, a ranking scheme was proposed to organize the results so that the communication overhead could be effectively reduced.

### 3) HETEROGENEOUS SEARCH

Heterogeneous search techniques mainly include semantic/ontology-based search and resource/service retrieval. In particular, ontologies represent the concepts, types, and relationships of different domains [70]. Merging semantic and ontology mechanisms can assist the system to build a domain, task, and method combination search system. On the other hand, all the data and IoT devices are able to abstract resources and provide different services. To discover the resources and identify the associated services is the main approach for resource/service retrieval.

For example, Cassar *et al.* [71] proposed a hybrid semantic service matchmaking method, which utilizes latent semantic analysis. It analyzes the semantics and assigns weights to the different content by logical signature. The proposed scheme overcomes the critical challenge for semantic service matchmakers, which is the issue of synonymy. The scheme was also shown to increase the accuracy of searching synonymous content. Likewise, Fredj *et al.* [72] proposed the Semantic Web technologies RDF (an open semantic data format) and SPARQL (a query language for RDF-encoded data) schemes to search and obtain real-time content from IoT devices. Specifically, the proposed schemes encode the sensor data to RDF triplets, and can be utilized as triplets by SPARQL to enable searching.

In addition, Ding *et al.* [73] proposed a hybrid real-time search engine framework to leverage IoT resources based on spatial-temporal, value, and keyword search to identify resources. In detail, a Moving Object Grid-Sketched Spatial-Temporal R-Tree (MOGSSTR-Tree) is designed and

utilized to monitor the spatial-temporal attributes of the IoT resources. The MOGSSTR leverages the actual sampling values to trace the variety of IoT resources instead of trajectory units. It also increases the IoT searching speed and reduces the cost of the searching process. Meanwhile, Shemshadi *et al.* [20] created and abstracted two types of interfaces as resource search tools. These gather IoT datasets from real-time web-based maps. Additionally, Nunes *et al.* [74] proposed a Visual Search for Internet of Things (ViSIoT) platform to pull IoT data from a central repository. Their ViSIoT then transforms the data into a generic format, in order to support heterogeneous devices.

## IV. PROBLEM SPACE OF IoT SEARCH ENGINE

In this section, we propose and define a problem space of IoT search engines. To do so, we first introduce CPS and the relationship between CPS and IoT, in order to provide a better view of why IoT search services are necessary. Then, we propose the problem space for IoT search engines. Finally, based on the presented problem space, we outline opportunities and challenges for future research and development.

### A. CPS AND IoT

We now discuss the relationship between CPS and IoT, and propose our view, specifically that CPS have needs that require IoT search services, and that CPS automation is a critically important motivator for developing IoT search engines.

Generally speaking, CPS are the effective vertical integration of computation, communication, and command and control, consisting of a computation core (centralized or distributed computing), network system (wired or wireless network), physical components (sensors or actuators), and control applications [6]. CPS can be generalized as closed loop processes, otherwise known as feedback loops, in which sensors collect the related data, the computation core analyzes the data, and results and decisions are disseminated to actuators to enact control of the system. Fig. 4 shows the relationship between IoT and CPS. Since IoT is a horizontal network system that connects all the physical objects and smart devices, IoT can be considered as the network framework of the CPS and is located in the communication layer of the CPS framework. The figure clearly shows how applications and sensors/actuators of CPS communicate with each other via IoT.

Benefiting from the development of IoT, numerous CPS are now deployed in various fields to improve system performance. For instance, manufacturing CPS are able to control intelligent machines to produce and assemble products [75]. In addition, in smart transportation systems, sensors collect traffic information and connected applications can compute the most efficient routes to destinations [49]. Also, in the smart grid system, smart meters are deployed to monitor the electricity consumption in real-time. Based on the collected data, the smart grid can adjust the power generation to balance demand and supply [76]. In addition, based on the collected data, the smart grid can guide the electricity price to balance

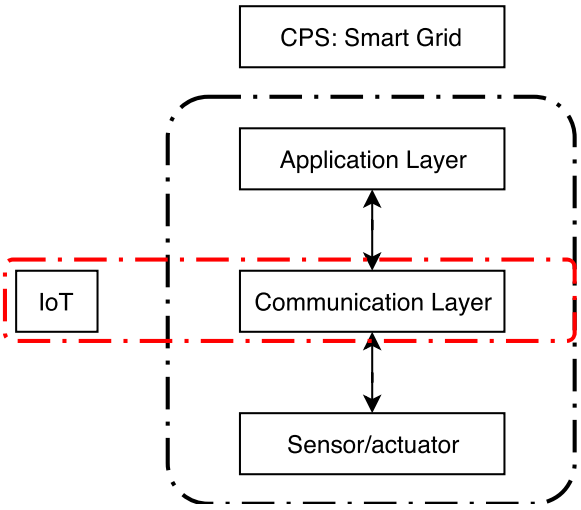


**FIGURE 4.** IoT vs. CPS.

demand and supply so that effective power use can be provided [34]. Moreover, in smart home systems, sensors are deployed throughout homes to collect and send information to servers for further analysis. Based on the data analysis results, smart home systems are able to obtain the owners' daily routines, such as work schedule, peak electricity usage, and others [77].

Furthermore, smart home systems are typically widely deployed within an area to collect data from the entire region. Meanwhile, the collected datasets can be sent to smart grid systems and utilized as the features of the training datasets for the smart grid system models. The smart grid systems are then able to adjust the power supply to match the electricity peaks based on the real, individualized data applied to achieve more comprehensive analysis results. In such situations, data acquisition and analysis are the keys to controlling the system and providing services. However, the data is usually stored in different CPS independently. For example, in the previous case, the users' routines are stored in the smart home system and the smart grid system cannot obtain the data directly. Therefore, IoT search is a feasible approach to solving data availability problems in CPS.

### B. PROBLEM SPACE

Fig. 5 shows the problem space of the IoT search engine. We generalize the problem space of IoT search engines into three dimensions. Here, the "x" axis represents the Query Types, which are divided into "Human" queries and "Machine" (i.e., "IoT device") queries. In IoT, both humans and machines have need to query datasets and IoT resources, one of the primary differences between web search and IoT search.

The "y" axis represents the properties, including Quality of Service (QoS) (e.g., search accuracy, speed) and security, arranged in a progressive manner. The basic QoS requirement is search accuracy, which is the first level requirement of QoS. Next, being able to achieve accurate searches, search speed

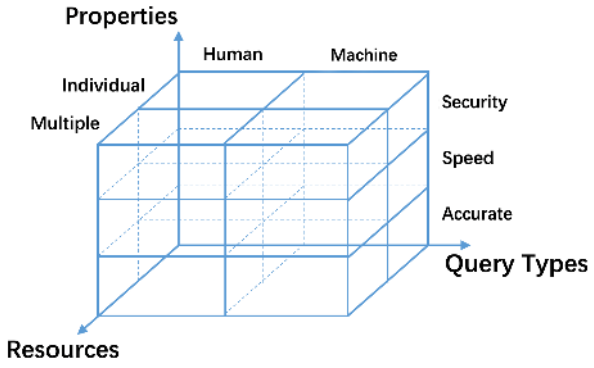


**FIGURE 5.** Problem space of the IoT search engines.

is crucial, as IoT search is often time-sensitive and the data is dynamic. Lastly, with accurate and rapid search achieved, security is a critical requirement of IoT search. In IoT search, adversaries may be able to insert tampered datasets in place of normal ones in order to attack the CPS and force decision errors [47], [78]. Thus, especially in the context of critical infrastructures, security is dire.

Finally, the ''z'' axis represents the resources, where ''Individual'' represents the search engine retrieving data from only one CPS repository, while ''Multiple'' indicates the search can retrieve data from multiple CPS repositories. There are a number of existing CPS that provide searching services. In the case of ''individual'' search services, in smart grid systems, users are able to search for electricity usage by finding the smart meter data for the specific area. However, the users are not able to obtain the temperature information from the smart grid systems at the same time, since the temperature information is stored in another CPS. On the other hand, some other CPS provide ''Multiple'' search services. For example, service provided by Uber is a typical CPS-based application that is used for providing peer-to-peer ridesharing services [79], [80]. For example, Uber not only shows the available vehicles, but also shows the route from customer to the destination, which is obtained from Google maps, and even provides road traffic information. Thus, integrating the multiple systems enables CPS to provide more comprehensive services.

### C. OPPORTUNITIES

Having identified the problem space of IoT search engines, we now present opportunities and challenges for IoT search engine research and development. In the following, we discuss the opportunities for IoT search engines from the perspectives of data retrieval, data comprehension and analysis, system automation, and artificial intelligence.

#### 1) DATA RETRIEVAL

With the increasing volume of IoT data and the increasing demand from data users, we find there to be an ideal opportunity to introduce novel IoT search engines and to improve IoT search for all IoT systems and CPS. Similar to web search engines, there are two key factors that boost the development of IoT search. The first is that data providers have massive valuable data, and the second is the increasing number of data users who seek to search and utilize IoT-related datasets. In this case, IoT search engines present an unprecedented opportunity and a large development space, supporting numerous smart-world applications. Currently, most IoT search engines are in the embryonic stage, with functional yet limited capabilities, such as the aforementioned Shodan and Thingful. Indeed, these existing search engines all have some limitations. For instance, Shodan can only search the information of IoT devices with open-APIs, while Thingful can only obtain static datasets that have been shared by the owners of the IoT devices. Therefore, there is an emerging demand to construct advanced IoT search engines with improved functionality.

#### 2) DATA COMPREHENSIVE ANALYSIS

As we discussed before, at the current stage, CPS operate individually, and cannot access data from any one unified application. However, in order to obtain more accurate results, the comprehensive analysis of multiple features and modalities of IoT data is necessary. For instance, in the event of a public safety emergency, emergency response departments and personnel require different types of data from critical infrastructure CPS [81], which include meteorological information, traffic information, energy supply information, wireless network information, etc. As the information is stored individually in each CPS, each must be accessed separately, and there are no IoT search portals able to search all the information by one query, leading to an obvious inefficiency. Likewise, in the smart city system, in order to obtain situation awareness of the entire city, the smart city system needs to communicate with many other CPS, including the traffic surveillance systems, meteorological information systems, vehicle tracking systems, etc., none of which are connected. Moreover, there are many critical issues that arise from connecting each individual CPS. Therefore, IoT search engines are required that can obtain various data from different CPS in one unified interface to increase data search efficiency.

#### 3) SYSTEM AUTOMATION

CPS assist humans in the monitoring and control of systems and infrastructure in real-time. Thus, data retrieval and analysis for dynamic data is key. As we discussed above, the automation of CPS is a closed loop process which includes system monitoring, data collection, data analysis, and execution. However, there are two problems that hinder the automation loop. One is the data search, which is currently hampered because all queries are sent by human users, meaning that the CPS require human interruption and intervention in the data collection stage. Therefore, the IoT search engines are required to respond to queries to/from both human users and machines (i.e., IoT devices) as necessary. Specifically, IoT devices should be able to send queries to the IoT search engines automatically and obtain the necessary

datasets for the data analysis process. The second problem is cross-system queries. As we mentioned above, CPS are executed independently. Thus, without the integration of multiple CPS, CPS cannot obtain the information stored in one another's repositories. Therefore, another opportunity arises for developing and leveraging IoT search engines.

#### 4) ARTIFICIAL INTELLIGENCE

Furthermore, new techniques and improving technologies, such as distributed computing (edge computing, edge/cloud combined) and machine learning [42], [46], [54], are highly integrated into CPS in order to endow intelligence to the system. Otherwise known as Artificial Intelligence (AI), systems that can make decisions autonomously based on data search and analysis, and those that can learn the requisite contexts and skills by themselves, are able to handle complex situations and tasks. For example, autonomous vehicles need to obtain relevant information without human intervention. First, the autonomous vehicles communicate with the IoT search engines to determine the destination. Then, the search engines search the traffic, weather, and parking lot information based on the route and return the information to the vehicles. Meanwhile, the vehicles send the query to the search engines to find the locations of gas or charging stations when the fuel/power supply is low [82]. Additionally, AI require more information and advanced data search services. Thus, the AI also have a high need of IoT search services, even more so than CPS automation [83].

### D. CHALLENGES

We now consider the challenges facing the development and improvement of IoT search engines. In particular, we classify the challenges into three categories: Dynamic Environments, Search Techniques, and Performance and Security.

#### 1) DYNAMIC ENVIRONMENTS

The IoT is composed of dynamic heterogeneous networks, which not only continuously generate high-velocity IoT data, but also change network structures frequently. This is a critical challenge for IoT search engines. In detail, the variability in the systems includes both dynamic data and dynamic network topology. IoT search engines need to crawl dynamic IoT data in real-time, while being able to react to network topology changes. Furthermore, scalability is also a challenge. As we mentioned above, IoT search engines should have the ability to extend their search area or be able to add new sensors and data repositories into the search area. However, in their current stage, IoT search engines all have some technical limitations to extending the search area. For instance, the Shodan IoT search engine can only search for IP address and hardware information of IoT devices, and cannot obtain such data without permission. Furthermore, while the Shodan search engine is reliable for dynamic datasets, the data update frequency is slow. In addition, IoT search engines need to consider the geographic location changes of mobile IoT devices, since users require the location of the specific IoT device. Finally, IoT search engines should be able to detect and react to the connectivity status of IoT devices, as the connectivity of IoT devices are not stable. Thus, there are numerous open issues that make it difficult for IoT search engines to crawl real-time data in dynamic environments.

#### 2) SEARCH TECHNIQUES

There are a number of technical challenges for IoT search engines in terms of search techniques. For instance, in responding to queries from both humans and machines (i.e., IoT devices), regarding queries from IoT devices, the number of instant queries will be very large. Thus, responding to a large number of queries in a very short time period is a challenge, especially at a rate not feasible to human users. In addition, providing efficient query processing and generating accurate and quick response to queries is critical [84]–[86]. For example, Wu *et al.* [84] proposed a time-series based framework using statistics-based techniques to carry out information aggregation of machine traffic in IoT systems, from which the aggregated search results provided insightful information for resource planning for IoT systems. Likewise, Quoc *et al.* [85] addressed the issue of query planning on spatio-temporal IoT data and proposed a scheme that leverages query similarity identification and machine learning techniques to improve query prediction accuracy.

Moreover, data acquisition is another problem for IoT search engines. Since the IoT data is updated frequently, conventional crawling algorithms will not be viable. New data acquisition algorithms are required to detect the dynamic data changes and collect the new data rapidly. In addition, the data acquisition process not only collects data from IoT devices, but also from user applications. Thus, data acquisition algorithms need to handle heterogeneous IoT data. Furthermore, accessing IoT resources, such as sensors, network devices, middlewares, computing platforms, and applications is a challenge. Because the IoT resources are managed by different organizations, the ability to obtain authorization to access the IoT resources is still unresolved. Finally, since the distributed computing platforms and machine learning-driven data analysis tools are integrated with IoT, how to integrate IoT search engines with those new techniques and tools to obtain better performance is another looming challenge.

#### 3) PERFORMANCE AND SECURITY

From the perspective of the performance of IoT search engines, challenges persist, as outlined in the problem space, above. First, since the IoT data is heterogeneous, it is difficult to accurately describe the specific data and specific IoT resources, and likewise, it will be difficult for the IoT search engine to identify and find. As a result, the quality of searching results will be hard to maintain. Thus, the accuracy of query results is a particular challenge. Second, the search speed is an important requirement for IoT systems. However, IoT resources are massive, and thus, how to satisfy the search speed requirement is a critical challenge. More importantly,

security is a challenge for all systems, including IoT search engines. In IoT search, adversaries could attack either the engines or the data sources in order to compromise the system. The IoT search engines involve numerous systems and excessive communications, potentially increasing the vulnerabilities of the system. For instance, adversaries may be able to easily inject the fake data by compromising the IoT sensors, because sensors are not well protected logically or physically (network vulnerability and physical environment risk). The injected false data can impact the search and analysis results to cause errors in the system, affecting smart-world systems such as smart grids, smart transportation, smart manufacturing, and others [47], [48], [78], [87]–[89]. In addition, adversaries may be able to directly attack the database on the search engines to compromise the searching results. To this end, comprehensive attack detection and protection strategies are still unresolved.

## V. FUTURE DIRECTIONS

In this section, we propose a vision for potential future research work and development directions for IoT search engines. Based on the survey and problem space developed, we summarize five major directions below.

### A. CO-DESIGN WITH OTHER TECHNIQUES

The first potential research direction is the integration of IoT search engines and search mechanisms with emerging, state-of-the-art, bleeding-edge network and computing techniques. With the ongoing development of distributed computing and machine learning technologies [42], [54], [55], the integration of these technologies with IoT is progressing. This integration should be extended to encompass search, as these emerging technologies can improve searching performance, resiliency, and security. Specifically, distributed computing, such as edge and fog computing, offloads computation tasks from cloud servers to edge computation nodes. Distributed computing has a hierarchical structure similar to IoT, and clearly shows potential for integration with IoT systems to provide low-latency computing and storage services [54], [55]. Since edge computation nodes are both topologically and geographically close to the collaborating IoT devices, data transmission and response times will be reduced, enabling real-time or near-real-time service. The distributed computing structures are able to solve problems in network delay, computing capacity limitations, and provide computation resource optimization.

In the case of IoT search, issues persist in leveraging distributed computing. These include, but are not limited to: (i) *Edge computing node deployment*, which is a critical issue for IoT search engines, as IoT is dynamic and devices may move during the data collection and computation process. Indeed, it is challenging for IoT devices to select optimal edge computing nodes normally, and the problem is made worse in the dynamic context, as mechanisms for switching between optimal edge computing nodes during movement are complex and the problem is not fully resolved. (ii) *Synchronization of distributed IoT search systems* is another ongoing challenge, which, given the computation and power limitations of nodes, requires that search tasks deployed in a distributed manner be efficient and resilient to services outage. Therefore, how to synchronize the search process on dispersed computing nodes is another promising research direction. (iii) *Research on network architectures and protocols* is still relatively limited. Edge computing nodes may be PCs, workstations, or mobile devices, and therefore, how to organize the heterogeneous networks with a number of heterogeneous devices is a challenge. (iv) *Next generation mobile networks*, including 5G, are the cellular infrastructures that may enable solutions to IoT connectivity problems [90], [91]. The 5G networks provide high-speed connections with low transmit energy cost, which creates a friendly ecosystem for IoT devices. Potential research directions include the leveraging of 5G to improve the performance of IoT search. For instance, one question is how to connect with IoT devices which use different protocols. In addition, Software Defined Networking (SDN) is a key feature of 5G, which provides flexible network structure and management [92]. SDN can help humans to easily manage IoT devices, but how to organize the IoT devices under SDN is another problem.

Similarly, machine learning has eclipsed other traditional big data analysis approaches, providing unique and impressive advantages in a variety of areas. For instance, machine learning algorithms can obtain more accurate results in large and dynamic datasets compared with other conventional data analysis methods. Furthermore, machine learning can be applied in a variety of ways, enabling accurate prediction, identification, and classification of data, actions, and tasks, and can be leveraged to realize AI. Thus, machine learning has the potential to assist IoT search in optimizing performance and analysis. However, there are several issues still remaining: (i) The training time for highly accurate machine learning models for production systems is unacceptable for time-sensitive IoT search, and especially for dynamic data. Thus, leveraging on-line learning strategies to handle dynamic data is a key research direction with implications for a variety of systems. (ii) The computation costs of machine learning models, especially in training, are non-trivial. Currently, the two major ways to reduce computational costs are data sampling to reduce total data throughput (and thus time), and data pre-preprocessing algorithms to reduce the training data sizes. While these are useful, it is not clear that these schemes are complete solutions, and new schemes to reduce the data needs of models would be highly beneficial. (iii) Additionally, optimizing machine learning algorithms for computation-limited devices that are available inside IoT systems is a further research direction with great potential, which already has a small body of work behind it, such as the optimization of trained models for resource constrained devices such as smartphones. Note that existing schemes are far from perfect solutions, and further work is needed.

### B. MULTI-SYSTEM INTERACTION

As we mentioned above, currently, CPS are built and managed independently. Thus, the integration of multiple CPS to enable interaction and inter-operation between different CPS is a possible research direction for IoT search engines. In the current stage of IoT search engine development, search servers generally serve applications of individual CPS management organizations, or at best serve multiple systems within a single organization. The data users have no choice but to install several different applications to complete the necessary data search and retrieval, which is a critical problem delaying the development of IoT search engines. The meta-search engine, as an emerging technology, sends queries to different search engines simultaneously in order to generate its own results. Thus, it is necessary to leverage the meta-search engine [93] concept to promote CPS integration and increase the interaction between different CPS. Generally speaking, there are two developmental phases or categories necessary for the integration of multiple CPS with meta-search, the first being hardware integration, and the second being software integration.

Hardware integration enables the connection of the different types of IoT devices anywhere and anytime, in order to respond to queries. Since IoT devices are diverse, there are a number of challenges and impediments to integrating the various IoT devices, including diverse network and communication protocols, hardware standards, geographic distance and location, and network connection. However, maintaining the interconnection of IoT devices is fundamental to IoT search. Thus, hardware integration should be a primary research direction to further enable IoT search.

Software integration includes the control, monitoring, management, and configuration of IoT devices through the utilization of a unified application or platform. As we discussed above, similar to web search engines, IoT search requires a unified portal. Since the data analysis requires comprehensive datasets from diverse CPS, the IoT search portal (single point search) is necessary. Moreover, the IoT devices must be able to send queries to the search engines to obtain related datasets from other CPS. Yet, the CPS belong to different organizations and their applications utilize different APIs, raising critical challenges to software integration. Utilizing middleware or a related federation layer between IoT search engines and applications is a potential solution to enable the necessary integration. The middleware can provide a unified API to enable both IoT search and other applications, and can enable IoT search engines to access data across different CPS repositories and nodes.

Generally speaking, after hardware and software integration, meta-search engines are able to send queries to all CPS and obtain data. Nonetheless, meta-search techniques raise some new challenges: (i) *Search Resource Selection:* Selecting the fastest search response servers is critical. Since there are massive numbers of servers distributed throughout the network to provide searching services, how to select the best server still needs significant research. Taking autonomous vehicles as an example, the meta-search engine sends queries to different search engines simultaneously. However, if the meta-search engine selects the search server whose response is slow, it will delay the searching results from being returned to the search engines and may result in accidents. (ii) *Data Format:* Due to different storage policies, the types of data collected from different IoT devices can be different (format, encoding, etc.). How to unify the different types of data formats into one is another pressing issue. (iii) *Access Authentication:* IoT search engines contain much more sensitive information than a typical web search engine. Thus, access authentication policies are an urgent demand for IoT search engines. How to build a trustworthy environment for interconnecting different entities is a challenging issue. Although web search engines indeed contain sensitive data such as user location, account, and email information, this information is traditionally freely generated by users (though perhaps unintentionally). In IoT search, we can consider the data to be more sensitive in some ways, and while some may be personal private data, we are also concerned about critical device information, and more, that may make IoT systems vulnerable to attack [89], [94], [95]. This information could be used to cause unallowable damage to individuals, notwithstanding the damage to system components and the system in total. If there is no access authentication in IoT search, all personal information might get leaked.

### C. PERFORMANCE OPTIMIZATION

Generally speaking, performance optimization is the inevitable approach to improve and maximize system efficacy. Similar to other systems, performance optimization is critical to improving the width and breadth of utilization and the availability of IoT search engines. Optimization approaches should consider, in particular, the time and space of search retrieval. Specifically, how to reduce search time and expand the search space are the main purposes of retrieval optimizations. As discussed in Section V-A, the integration of IoT search engines and other emerging technologies, such as distributed computing and machine learning-based data analytics, can play an important role in realizing IoT search capabilities. However, this integration raises additional questions, and how to overcome these new problems requires further research. For instance, we could leverage on-line machine learning to reduce the training time for machine learning models utilized in IoT search systems [96], and could apply distributed learning strategies (e.g., federated learning, transfer learning) to increase the scalability of machine learning to handle large volumes of training data [97], [98]. Yet, neither of these technologies are fully developed and optimized for use in the particular case of IoT search in scalability, efficiency and effectiveness.

At the same time, we should encourage data owners to share data publicly in order to extend the breadth of the search space and enable collaborative improvement of all systems. As further incentive, data owners have the potential to trade both physical commodities in the IoT system and

data via digital markets such that the commercial value of the data remains intact and can be recuperated [16], [99]–[102]. On one hand, for the physical world commodities, such as electricity in the smart grid system, the secondary market among microgrids in the smart grid allows entities to buy or sell energy efficiently. To tackle the security and privacy concerns, new schemes would be conceived to ensure not only economic properties via different trading algorithms, but also security and privacy properties via secure protocol design, multi-party computing and differential privacy [103], [104].

On the other hand, IoT big data is considered to be the key to unlocking the next great wave of growth in productivity, supporting numerous smart-world applications. With the exponential growth of data in IoT-based systems, how to efficiently utilize the data becomes a critical issue. This calls for the development of a big data market that enables efficient and secure data trading. By pushing data as a kind of commodity into a digital market, the data owners and consumers are able to connect with each other, sharing and further increasing the utility of the data. However, to enable such a market for data trading, several challenges need to be addressed: (i) how to determine the proper price for the data to be sold or purchased, (ii) how to design a trustworthy trading platform and schemes to enable the maximization of social welfare of trading participants with efficiency, security and privacy guarantees, and (iii) how to protect the traded data from being resold to maintain the value of the data. Future research must provide a clear and deep understanding of commodity and big data trading markets and initiate a scientific foundation for trading commodities and data in an efficient, secure and privacy-preserving manner. Note that pushing data to digital markets as a commodity can promote data sharing and extend the searching space. In addition, it is necessary to attend to the optimization of the IoT search engines themselves. As we discussed before, how to leverage and update crawling algorithms to suit IoT scenarios and IoT data with high vitality is a meaningful problem. Likewise, how to store and generate the index for different types of IoT data also needs to be addressed.

### D. SECURITY AND PRIVACY

The widespread adoption of IoT devices and heterogeneous network structures provide new vulnerabilities for adversaries and drastically increases the risks of attacks, in both breadth and depth. Indeed, a number of recent IoT-based attacks have exacerbated the urgency and need for assuring security in IoT [105]–[109], [109]–[111]. In the context of IoT search engines, attacks have the potential to induce catastrophic damage to data analysis results, affecting CPS and the security and privacy of individuals [89]. For instance, in the smart transportation system, attacks against On-Board Units (OBUs) and traffic lights can generate serious traffic congestion, impact traffic prediction, and potentially cause harm and human injury. In the smart city, adversaries could attack the city surveillance system and manipulate the directions of the cameras in order to escape tracking. In addition, in the smart grid system, adversaries could compromise smart meters and inject false data to impact power supply management. Furthermore, since smartphone and wearable devices are widespread, adversaries could attack such devices to obtain personal and private data, health information, intercept package delivery, or even track the user. How to design effective privacy protection schemes, including differential privacy [94], secure multi-part computing [99], [112], and anonymous communication [113], [114], in IoT search remain a challenging issue.

In general, attacks on IoT search engines can be launched to target service, query, and devices: (i) *Attacks on services:* Adversaries directly attacking the IoT search engines themselves have the potential to disable search services and deny queries being serviced. In this case, the CPS cannot obtain real-time data, negatively impacting the data analysis results. Erroneousness results may cause system failures, should no redundancies be in place. (ii) *Attacks on queries:* Adversaries may launch attacks on queries in the IoT search process. In conventional web search, humans can inspect the search results in order to verify the accuracy of the query and prevent tampered queries from impacting systems and decisions. However, in IoT search, many queries are sent by machines such as IoT devices, and attack detection schemes will be necessary to inspect the correctness of the queries. (iii) *Attacks on devices:* Adversaries have the potential to launch attacks on IoT devices, which are more critically vulnerable due to their limited computing resources and lack of security features. Since most IoT data is collected by IoT sensors, maliciously manipulating sensors can damage data and CPS at the source. In addition, new attack detection and protection strategies for IoT sensors are necessary to handle their limitations, which increase the risks of adversaries subverting IoT systems. All these approaches have the potential to disturb the performance of CPS and IoT search engines and cause irreparable damage. Therefore, for the IoT search engines and the betterment of IoT as a whole, a significant amount of research on security and privacy is necessary for the future.

### E. FOUNDATION FOR DATA AND NETWORK SCIENCE

Big Data has come to be a dominant force in our society, affecting all manner of industries, products, and livelihoods. Moreover, the foundational aspects of Big Data include networked systems, computation systems, cloud infrastructures, software and systems security, distributed systems and multiprocessing, data collection and storage, data analytics and assessment, and many more. As a logical extension of Big Data, IoT Search, or the search of distributed, dynamic big data, implicates all such foundational techniques and technologies, with the addition of algorithms, methodologies, and technologies particular to search, retrieval, and return. Thus, the study of IoT search and the development of IoT search engines offer clear opportunities for interested computer, network, and data science researchers and practitioners to

advance their knowledge and understanding of foundational computer, computation, data, and statistical sciences.

In the study of traditional foundational computational, computer, and data sciences, IoT search implicates aspects of big data analytics and assessment, machine learning, automation, command and control design and management, and so on. In particular, the hardware and software challenges for big data increase in scale under IoT search, as systems become more massively distributed, and data becomes more massively large and continuous. The many V's of Big Data (Value, Veracity, Volume, Velocity, Variety, etc.) are clearly realized, and must be accounted for. Additionally, the analysis of such data faces massively increased potential, as well as massively increased noise and vulnerability, along with the inability for complete data review and understanding. Thus, the need for astute design and understanding of the practicalities and limitations of a variety of combined technologies is imperative.

Given the reliance of all modern societies on the Internet and networked systems, it is imperative that these systems be understood, improved, and innovated on to better ensure the security and reliability of users, their data, and the systems that we all rely upon. Necessarily implicating networked systems and the Internet, IoT search requires proper consideration for heterogeneous network conditions, and will require advancements in cloud and edge technologies, machine-to-machine communication techniques and infrastructures, 5G, MANET, SDN, and others. Not simply an improvement in service, networks to support IoT search must contend with the massively distributed IoT, aggregation of massive sensor data, the service provisioning for critical versus non-critical applications and hardware, and so on. Future work toward improving networks, as big communication infrastructure, from all aspects, including throughput, bandwidth, organization, optimization, function, and security, are necessary to realize IoT search engine capabilities. Thus, advances in techniques, technologies, algorithms, design, etc. achieved through targeted and collaborative research are necessary and imperative, both overall, and specifically to realize IoT search capabilities.

## VI. FINAL REMARKS

With the development of IoT sensor networks and powerful emerging technologies, such as distributed computing and machine learning, there is an urgent demand to develop IoT search engines to retrieve IoT data, an extension of big data that has real-world implications. In order to provide a general view of IoT search engine concepts, research, and progress, in this work we surveyed IoT search and search engine techniques. Primarily, we have focused on the development progress of IoT search engines, first briefly reviewing the development of conventional web search engines as a representative example. We also studied the major techniques of web search, and systematically studied existing IoT search techniques, discussing the differences between traditional web and novel IoT search engines. Based on our study, we developed the problem space for IoT search engines and discussed opportunities and challenges that remain. Finally, we proposed a vision for future research and highlighted key areas in which IoT search and IoT technologies in general must progress to ensure more reliable, safe, and secure distributed sensing and computing systems, supporting numerous emergent smart-world applications.

## REFERENCES

[1] *Shodan*. Accessed: Jun. 13, 2019. [Online]. Available: https://www.shodan.io/
[2] *Thingful*. Accessed: Jun. 13, 2019. [Online]. Available: https://www.thingful.net/
[3] *Censys*. Accessed: Jun. 13, 2019. [Online]. Available: https://censys.io/
[4] *Reposify*. Accessed: Jun. 13, 2019. [Online]. Available: https://www.reposify.com/
[5] J. A. Stankovic, "Research directions for the Internet of Things," *IEEE Internet Things J.*, vol. 1, no. 1, pp. 3–9, Feb. 2014.
[6] J. Lin, W. Yu, N. Zhang, X. Yang, H. Zhang, and W. Zhao, "A survey on Internet of things: Architecture, enabling technologies, security and privacy, and applications," *IEEE Internet Things J.*, vol. 4, no. 5, pp. 1125–1142, Oct. 2017.
[7] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of Things for smart cities," *IEEE Internet Things J.*, vol. 1, no. 1, pp. 22–32, Feb. 2014.
[8] S. M. Riazul Islam, D. Kwak, M. Humaun Kabir, M. Hossain, and K.-S. Kwak, "The Internet of Things for health care: A comprehensive survey," *IEEE Access*, vol. 3, pp. 678–708, Jun. 2015.
[9] L. Da Xu, W. He, and S. Li, "Internet of Things in industries: A survey," *IEEE Trans. Ind. Informat.*, vol. 10, no. 4, pp. 2233–2243, Nov. 2014.
[10] M. Aazam, K. A. Harras, and S. Zeadally, "Fog computing for 5G tactile industrial Internet of Things: QoE-aware resource allocation model," *IEEE Trans. Ind. Informat.*, vol. 15, no. 5, pp. 3085–3092, May 2019.
[11] S. B. Baker, W. Xiang, and I. Atkinson, " Internet of Things for smart healthcare: Technologies, challenges, and opportunities," *IEEE Access*, vol. 5, pp. 26521–26544, 2017.
[12] Xiaomi Team. *Xiaomi and IKEA Partner to Bring Smart Connected Homes to More Users*. [Online]. Available: https://bit.ly/2GnJWWy
[13] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," *Comput. Netw.*, vol. 54, no. 15, pp. 2787–2805, Oct. 2010.
[14] Y. Liu, Y. He, M. Li, J. Wang, K. Liu, and X. Li, "Does wireless sensor network scale? A measurement study on GreenOrbs," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 10, pp. 1983–1993, Oct. 2013.
[15] P. Barnaghi and A. Sheth, "On searching the Internet of Things: Requirements and challenges," *IEEE Intell. Syst.*, vol. 31, no. 6, pp. 71–75, Nov./Dec. 2016.
[16] F. Liang, W. Yu, D. An, Q. Yang, X. Fu, and W. Zhao, "A survey on big data market: Pricing, trading and protection," *IEEE Access*, vol. 6, pp. 15132–15154, 2018.
[17] K. Romer, B. Ostermaier, F. Mattern, M. Fahrmair, and W. Kellerer, "Real-time search for real-world entities: A survey," *Proc. IEEE*, vol. 98, no. 11, pp. 1887–1902, Nov. 2010.
[18] D. Zhang, L. T. Yang, and H. Huang, "Searching in Internet of Things: Vision and challenges," in *Proc. IEEE 9th Int. Symp. Parallel Distrib. Process. Appl.*, May 2011, pp. 201–206.
[19] StatCounter. *Search Engine Market Share Worldwide*. [Online]. Available: http://gs.statcounter.com/search-engine-market-share
[20] A. Shemshadi, Q. Z. Sheng, and Y. Qin, "ThingSeek: A crawler and search engine for the Internet of Things," in *Proc. 39th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, 2016, pp. 1149–1152.
[21] J. Bar-Ilan, M. Levene, and M. Mat-Hassan, "Methods for evaluating dynamic changes in search engine rankings: A case study," *J. Document.*, vol. 62, no. 6, pp. 708–729, 2006.
[22] C. D. Nguyen, "Smart search engine," U.S. Patent 14 455 482, Feb. 11 2016.
[23] J. Boyan, D. Freitag, and T. Joachims, "A machine learning architecture for optimizing Web search engines," in *Proc. AAAI Workshop Internet Based Inf. Syst.*, 1996, pp. 1–8.
[24] *StatCounter*. Accessed: Jun. 15, 2019. [Online]. Available: http://gs.statcounter.com/search-engine-market-share/

[25] C. Manning, P. Raghavan, and H. Schütze, "Introduction to information retrieval," *Natural Lang. Eng.*, vol. 16, no. 1, pp. 100–103, 2010.

[26] S. Brin and L. Page, "The anatomy of a large-scale hypertextual Web search engine," *Comput. Netw. ISDN Syst.*, vol. 30, nos. 1–7, pp. 107–117, Apr. 1998.

[27] A. N. Langville and C. D. Meyer, *Google's PageRank and Beyond: The Science of Search Engine Rankings*. Princeton, NJ, USA: Princeton Univ. Press, 2011.

[28] StatCounter. *Search Engine Market Share Worldwide*. Accessed: Jun. 15, 2019. [Online]. Available: http://gs.statcounter.com/search-engine-market-share

[29] S. Ghemawat, H. Gobioff, and S.-T. Leung, "The Google file system," *ACM SIGOPS Oper. Syst. Rev.*, vol. 37, no. 5, pp. 29–43, Dec. 2003.

[30] J. Dean and S. Ghemawat, "MapReduce: Simplified data processing on large clusters," *Commun. ACM*, vol. 51, no. 1, pp. 107–113, 2008.

[31] F. Chang, J. Dean, S. Ghemawat, W. C. Hsieh, D. A. Wallach, M. Burrows, T. Chandra, A. Fikes, and R. E. Gruber, "Bigtable: A distributed storage system for structured data," *ACM Trans. Comput. Syst.*, vol. 26, no. 2, pp. 4:1–4:26, Jun. 2008.

[32] E. Schmidt and J. Rosenberg, *How Google Works*. London, U.K.: Hachette, 2014.

[33] M. Chen, S. Mao, and Y. Liu, "Big data: A survey," *Mobile Netw. Appl.*, vol. 19, no. 2, pp. 171–209, Apr. 2014.

[34] J. Lin, W. Yu, and X. Yang, "Towards multistep electricity prices in smart grid electricity markets," *IEEE Trans. Parallel Distrib. Syst.*, vol. 27, no. 1, pp. 286–302, Jan. 2016.

[35] X. Fang, S. Misra, G. Xue, and D. Yang, "Smart grid—The new and improved power grid: A survey," *IEEE Commun. Surveys Tuts.*, vol. 14, no. 4, pp. 944–980, 4th Quart., 2012.

[36] Y. Zhang, Q. Yang, W. Yu, D. An, D. Li, and W. Zhao, "An online continuous progressive second price auction for electric vehicle charging," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 2907–2921, Apr. 2019.

[37] J. Li, X. Sun, Q. Liu, W. Zheng, H. Liu, and J. A. Stankovic, "Planning electric vehicle charging stations based on user charging behavior," in *Proc. IEEE/ACM 3rd Int. Conf. Internet-Things Design Implement. (IoTDI)*, Apr. 2018, pp. 225–236.

[38] T. Anagnostopoulos, A. Zaslavsky, K. Kolomvatsos, A. Medvedev, P. Amirian, J. Morley, and S. Hadjieftymiades, "Challenges and opportunities of waste management in IoT-enabled smart cities: A survey," *IEEE Trans. Sustain. Comput.*, vol. 2, no. 3, pp. 275–289, Jul. 2017.

[39] S. Mallapuram, N. Ngwum, F. Yuan, C. Lu, and W. Yu, "Smart city: The state of the art, datasets, and evaluation platforms," in *Proc. IEEE/ACIS 16th Int. Conf. Comput. Inf. Sci. (ICIS)*, May 2017, pp. 447–452.

[40] G. Fayard, "Road injury prevention in China: Current state and future challenges," *J. Public Health Policy*, vol. 40, no. 1, pp. 1–16, 2019.

[41] J. C. F. de Winter, D. Dodou, R. Happee, and Y. B. Eisma, "Will vehicle data be shared to address the how, where, and who of traffic accidents?," *Eur. J. Futures Res.*, vol. 7, no. 1, p. 2, 2019.

[42] W. G. Hatcher and W. Yu, "A survey of deep learning: Platforms, applications and emerging research trends," *IEEE Access*, vol. 6, pp. 24411–24432, 2018.

[43] O. Elijah, T. A. Rahman, I. Orikumhi, C. Y. Leow, and M. N. Hindia, "An overview of Internet of Things (IoT) and data analytics in agriculture: Benefits and challenges," *IEEE Internet Things J.*, vol. 5, no. 5, pp. 3758–3773, Oct. 2018.

[44] M. Mohammadi, A. Al-Fuqaha, S. Sorour, and M. Guizani, "Deep learning for IoT big data and streaming analytics: A survey," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 4, pp. 2923–2960, 4th Quart., 2018.

[45] E. Hossain, I. Khan, F. Un-Noor, S. S. Sikander, and M. S. H. Sunny, "Application of big data and machine learning in smart grid, and associated security concerns: A review," *IEEE Access*, vol. 7, pp. 13960–13988, 2019.

[46] H. Li, K. Ota, and M. Dong, "Learning IoT in edge: Deep learning for the Internet of things with edge computing," *IEEE Netw.*, vol. 32, no. 1, pp. 96–101, Jan. 2018.

[47] Q. Yang, J. Yang, W. Yu, D. An, N. Zhang, and W. Zhao, "On false data-injection attacks against power system state estimation: Modeling and countermeasures," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 3, pp. 717–729, Mar. 2014.

[48] Y. Liu, P. Ning, and M. K. Reiter, "False data injection attacks against state estimation in electric power grids," in *Proc. 16th ACM Conf. Comput. Commun. Secur. (CCS)*. New York, NY, USA: ACM, 2009, pp. 21–32. doi: 10.1145/1653662.1653666.

[49] J. Lin, W. Yu, X. Yang, Q. Yang, X. Fu, and W. Zhao, "A novel dynamic en-route decision real-time route guidance scheme in intelligent transportation systems," in *Proc. IEEE 35th Int. Conf. Distrib. Comput. Syst.*, Jun./Jul. 2015, pp. 61–72.

[50] Y. Lv, Y. Duan, W. Kang, Z. Li, and F.-Y. Wang, "Traffic flow prediction with big data: A deep learning approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 2, pp. 865–873, Apr. 2015.

[51] J. Contreras-Castillo, S. Zeadally, and J. A. Guerrero-Ibañez, "Internet of vehicles: Architecture, protocols, and security," *IEEE Internet Things J.*, vol. 5, no. 5, pp. 3701–3709, Oct. 2018.

[52] 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, 2016.

[53] H. Ma and W. Liu, "A progressive search paradigm for the Internet of things," *IEEE MultiMedia*, vol. 25, no. 1, pp. 76–86, Jan./Mar. 2018.

[54] W. Yu, F. Liang, X. He, W. G. Hatcher, C. Lu, J. Lin, and X. Yang, "A survey on the edge computing for the Internet of Things," *IEEE Access*, vol. 6, pp. 6900–6919, 2018.

[55] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet Things J.*, vol. 3, no. 5, pp. 637–646, Oct. 2016.

[56] C. Mouradian, D. Naboulsi, S. Yangui, R. H. Glitho, M. J. Morrow, and P. A. Polakos, "A comprehensive survey on fog computing: State-of-the-art and research challenges," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 1, pp. 416–464, 1st Quart., 2018.

[57] M. A. Razzaque, M. Milojevic-Jevric, A. Palade, and S. Clarke, "Middleware for Internet of Things: A survey," *IEEE Internet Things J.*, vol. 3, no. 1, pp. 70–95, Feb. 2016.

[58] S. H. L. Liang and C.-Y. Huang, " GeoCENS: A geospatial cyberinfrastructure for the world-wide sensor Web," *Sensors*, vol. 13, no. 10, pp. 13402–13424, 2013.

[59] S. Mayer, D. Guinard, and V. Trifa, "Searching in a Web-based infrastructure for smart things," in *Proc. 3rd IEEE Int. Conf. Internet Things*, Oct. 2012, pp. 119–126.

[60] W. Wang, S. De, G. Cassar, and K. Moessner, "An experimental study on geospatial indexing for sensor service discovery," *Expert Syst. Appl.*, vol. 42, no. 7, pp. 3528–3538, May 2015.

[61] Y. Fathy, P. Barnaghi, S. Enshaeifar, and R. Tafazolli, "A distributed in-network indexing mechanism for the Internet of Things," in *Proc. IEEE 3rd World Forum Internet Things (WF-IoT)*, Dec. 2016, pp. 585–590.

[62] S. Samtani, S. Yu, H. Zhu, M. Patton, J. Matherly, and H. Chen, "Identifying supervisory control and data acquisition (SCADA) devices and their vulnerabilities on the Internet of Things (IoT): A text mining approach," *IEEE Intell. Syst.*, to be published.

[63] M. Blackstock and R. Lea, "IoT mashups with the WoTKit," in *Proc. 3rd IEEE Int. Conf. Internet Things*, Oct. 2012, pp. 159–166.

[64] J. Franks, P. Hallam-Baker, J. Hostetler, S. Lawrence, P. Leach, A. Luotonen, and L. Stewart, *HTTP Authentication: Basic and Digest Access Authentication*, document RFC 2617, 1999.

[65] C. Truong and K. Römer, "Content-based sensor search for the Web of things," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2013, pp. 2654–2660.

[66] W. T. Lunardi, E. de Matos, R. Tiburski, L. A. Amaral, S. Marczak, and F. Hessel, "Context-based search engine for industrial IoT: Discovery, search, selection, and usage of devices," in *Proc. IEEE 20th Conf. Emerg. Technol. Factory Autom. (ETFA)*, Sep. 2015, pp. 1–8.

[67] B. M. Elahi, K. Romer, B. Ostermaier, M. Fahrmair, and W. Kellerer, "Sensor ranking: A primitive for efficient content-based sensor search," in *Proc. Int. Conf. Inf. Process. Sensor Netw.*, Apr. 2009, pp. 217–228.

[68] P. Zhang, Y.-A. Liu, F. Wu, and B. Tang, "Matching state estimation scheme for content-based sensor search in the Web of things," *Int. J. Distrib. Sensor Netw.*, vol. 11, no. 11, 2015, Art. no. 326780.

[69] P. Zhang, Y. Liu, F. Wu, S. Liu, and B. Tang, "Low-overhead and high-precision prediction model for content-based sensor search in the Internet of Things," *IEEE Commun. Lett.*, vol. 20, no. 4, pp. 720–723, Apr. 2016.

[70] S. Pattar, R. Buyya, K. Venugopal, S. S. Iyengar, and L. M. Patnaik, "Searching for the IoT resources: Fundamentals, requirements, comprehensive review, and future directions," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 3, pp. 2101–2132, 3rd Quart., 2018.

[71] G. Cassar, P. Barnaghi, W. Wang, and K. Moessner, "A hybrid semantic matchmaker for IoT services," in *Proc. IEEE Int. Conf. Green Comput. Commun.*, Nov. 2012, pp. 210–216.

[72] S. Ben Fredj, M. Boussard, D. Kofman, and L. Noirie, "Efficient semantic-based IoT service discovery mechanism for dynamic environments," in *Proc. IEEE 25th Annu. Int. Symp. Pers., Indoor, Mobile Radio Commun. (PIMRC)*, Sep. 2014, pp. 2088–2092.
[73] 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, 2014.
[74] L. Nunes, J. Estrella, L. Nakamura, R. de Libardi, C. Ferreira, L. Jorge, C. Perera, and S. Reiff-Marganiec, "A distributed sensor data search platform for Internet of things environments," 2016, *arXiv:1606.07932*. [Online]. Available: https://arxiv.org/abs/1606.07932
[75] H. Xu, W. Yu, D. Griffith, and N. Golmie, "A survey on industrial Internet of Things: A cyber-physical systems perspective," *IEEE Access*, vol. 6, pp. 78238–78259, 2018.
[76] P. Moulema, W. Yu, D. Griffith, and N. Golmie, "On effectiveness of smart grid applications using co-simulation," in *Proc. 24th Int. Conf. Comput. Commun. Netw. (ICCCN)*, Aug. 2015, pp. 1–8.
[77] P. Siano, "Demand response and smart grids—A survey," *Renew. Sustain. Energy Rev.*, vol. 30, pp. 461–478, Feb. 2014.
[78] J. Lin, W. Yu, N. Zhang, X. Yang, and L. Ge, "Data integrity attacks against dynamic route guidance in transportation-based cyber-physical systems: Modeling, analysis, and defense," *IEEE Trans. Veh. Technol.*, vol. 67, no. 9, pp. 8738–8753, Sep. 2018.
[79] M. Zhu, X. Liu, and X. Wang, "An online ride-sharing path-planning strategy for public vehicle systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 2, pp. 616–627, Feb. 2019.
[80] M. Zhu, X.-Y. Liu, F. Tang, M. Qiu, R. Shen, W. Shu, and M.-Y. Wu, "Public vehicles for future urban transportation," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 12, pp. 3344–3353, Dec. 2016.
[81] W. Yu, H. Xu, J. Nguyen, E. Blasch, A. Hematian, and W. Gao, "Survey of public safety communications: User-side and network-side solutions and future directions," *IEEE Access*, vol. 6, pp. 70397–70425, 2018.
[82] M. Gerla, E.-K. Lee, G. Pau, and U. Lee, "Internet of vehicles: From intelligent grid to autonomous cars and vehicular clouds," in *Proc. IEEE World Forum Internet Things (WF-IoT)*, Mar. 2014, pp. 241–246.
[83] S. Madakam, R. Ramaswamy, and S. Tripathi, "Internet of Things (IoT): A literature review," *J. Comput. Commun.*, vol. 3, no. 5, p. 164, 2015.
[84] Y. Wu, Y. Cui, W. Yu, C. Lu, and W. Zhao, "Modeling and forecasting of timescale network traffic dynamics in M2M communications," in *Proc. 39th IEEE Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Jul. 2019, pp. 711–721.
[85] H. N. M. Quoc, M. Serrano, J. G. Breslin, and D. L. Phuoc, "A learning approach for query planning on spatio-temporal IoT data," in *Proc. 8th Int. Conf. Internet Things (IOT)*. New York, NY, USA: ACM, 2018, pp. 1:1–1:8. [Online]. Available: http://doi.acm.org/10.1145/3277593.3277598
[86] P. W. Widya, Y. Yustiawan, and J. Kwon, "A oneM2M-based query engine for Internet of Things (IoT) data streams," *Sensors*, vol. 18, no. 10, 2018. [Online]. Available: https://www.mdpi.com/1424-8220/18/10/3253
[87] Q. Yang, D. Li, W. Yu, Y. Liu, D. An, X. Yang, and J. Lin, "Toward data integrity attacks against optimal power flow in smart grid," *IEEE Internet Things J.*, vol. 4, no. 5, pp. 1726–1738, Oct. 2017.
[88] N. Tuptuk and S. Hailes, "Security of smart manufacturing systems," *J. Manuf. Syst.*, vol. 47, pp. 93–106, Apr. 2018.
[89] X. Liu, C. Qian, W. G. Hatcher, H. Xu, W. Liao, and W. Yu, "Secure Internet of Things (IoT)-based smart-world critical infrastructures: Survey, case study and research opportunities," *IEEE Access*, vol. 7, pp. 79523–79544, 2019.
[90] M. Agiwal, A. Roy, and N. Saxena, "Next generation 5G wireless networks: A comprehensive survey," *IEEE Commun. Surveys Tut.*, vol. 18, no. 3, pp. 1617–1655, 3rd Quart., 2016.
[91] W. Yu, H. Xu, H. Zhang, D. Griffith, and N. Golmie, "Ultra-dense networks: Survey of state of the art and future directions," in *Proc. 25th Int. Conf. Comput. Commun. Netw. (ICCCN)*, Aug. 2016, pp. 1–10.
[92] D. Kreutz, F. Ramos, P. E. Veríssimo, C. E. Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-defined networking: A comprehensive survey," *Proc. IEEE*, vol. 103, no. 1, pp. 14–76, Jan. 2015.
[93] A. Halavais, *Search Engine Society*. Hoboken, NJ, USA: Wiley, 2017.
[94] X. Yang, T. Wang, X. Ren, and W. Yu, "Survey on improving data utility in differentially private sequential data publishing," *IEEE Trans. Big Data*, to be published.
[95] D. Li, Q. Yang, D. An, W. Yu, X. Yang, and X. Fu, "On location privacy-preserving online double auction for electric vehicles in microgrids," *IEEE Internet Things J.*, to be published.
[96] F. Liang, W. G. Hatcher, G. Xu, W. Liao, and W. Yu, "Towards online deep learning based energy forecasting," in *Proc. IEEE Int. Conf. Comput. Commun. Netw. (ICCCN)*, Jul. 2019.
[97] Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated machine learning: Concept and applications," *ACM Trans. Intell. Syst. Technol.*, vol. 10, no. 2, pp. 12:1–12:19, 2019. doi: 10.1145/3298981.
[98] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.
[99] W. Gao, W. Yu, F. Liang, W. G. Hatcher, and C. Lu, "Privacy-preserving auction for big data trading using homomorphic encryption," *IEEE Trans. Netw. Sci. Eng.*, to be published.
[100] D. An, Q. Yang, W. Yu, D. Li, and Y. Zhang, "Towards truthful auction for big data trading," in *Proc. 36th IEEE Int. Perform. Comput. Commun. Conf. (IPCCC)*, Dec. 2017, pp. 1–7.
[101] Y. Jiao, P. Wang, D. Niyato, M. A. Alsheikh, and S. Feng, "Profit maximization auction and data management in big data markets," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, May 2017, pp. 1–6.
[102] D. Niyato, D. T. Hoang, N. C. Luong, P. Wang, D. I. Kim, and Z. Han, "Smart data pricing models for the Internet of Things: A bundling strategy approach," *IEEE Netw.*, vol. 30, no. 2, pp. 18–25, Feb. 2016.
[103] D. An, Q. Yang, W. Yu, X. Yang, X. Fu, and W. Zhao, "SODA: Strategy-proof online double auction scheme for multimicrogrids bidding," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 48, no. 7, pp. 1177–1190, Jul. 2018.
[104] D. Li, Q. Yang, W. Yu, D. An, X. Yang, and W. Zhao, "A strategy-proof privacy-preserving double auction mechanism for electrical vehicles demand response in microgrids," in *Proc. 36th IEEE Int. Perform. Comput. Commun. Conf. (IPCCC)*, Dec. 2017, pp. 1–8.
[105] M. Antonakakis, T. April, M. Bailey, M. Bernhard, E. Bursztein, J. Cochran, Z. Durumeric, J. A. Halderman, L. Invernizzi, M. Kallitsis, and D. Kumar, "Understanding the mirai botnet," in *Proc. 26th USENIX Conf. Secur. Symp. (SEC)*. Berkeley, CA, USA: USENIX Association, 2017, pp. 1093–1110. [Online]. Available: http://dl.acm.org/citation.cfm?id=3241189.3241275
[106] S. Soltan, P. Mittal, and H. V. Poor, "BlackIoT: IoT botnet of high wattage devices can disrupt the power grid," in *Proc. 27th USENIX Conf. Secur. Symp. (SEC)*. Berkeley, CA, USA: USENIX Association, 2018, pp. 15–32. [Online]. Available: http://dl.acm.org/citation.cfm?id=3277203.3277206
[107] Z. Ling, J. Luo, Y. Xu, C. Gao, K. Wu, and X. Fu, "Security vulnerabilities of Internet of Things: A case study of the smart plug system," *IEEE Internet Things J.*, vol. 4, no. 6, pp. 1899–1909, Dec. 2017.
[108] E. Bertino and N. Islam, "Botnets and Internet of Things security," *Computer*, vol. 50, no. 2, pp. 76–79, Feb. 2017. doi: 10.1109/MC.2017.62.
[109] A. Morse. (May 2017). Investigation: WannaCry cyber attack and the NHS. National Audit Office. [Online]. Available: https://www.nao.org.uk/wp-content/uploads/2017/10/Investigation-WannaCry-cyber-attack-and-the-NHS.pdf
[110] R. Williams, E. McMahon, S. Samtani, M. Patton, and H. Chen, "Identifying vulnerabilities of consumer Internet of Things (IoT) devices: A scalable approach," in *Proc. IEEE Int. Conf. Intell. Secur. Inform. (ISI)*, Jul. 2017, pp. 179–181.
[111] B. Pearson, L. Luo, Y. Zhang, R. Dey, Z. Ling, M. Bassiouni, and X. Fu, "On misconception of hardware and cost in IoT security and privacy," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2019, pp. 1–7.
[112] D. W. Archer, D. Bogdanov, Y. Lindell, L. Kamm, K. Nielsen, J. I. Pagter, N. P. Smart, and R. N. Wright, "From keys to databases—Real-world applications of secure multi-party computation," *Comput. J.*, vol. 61, no. 12, pp. 1749–1771, Dec. 2018.
[113] *Tor*. Accessed: Jun. 17, 2019. [Online]. Available: https://www.torproject.org/
[114] W. Yu, X. Fu, S. Graham, D. Xuan, and W. Zhao, "DSSS-based flow marking technique for invisible traceback," in *Proc. IEEE Symp. Security Privacy (SP)*, May 2007, pp. 18–32.

**FAN LIANG** received the bachelor's degree in computer science from Northwestern Polytechnical University, China, in 2005, and the master's degree in computer engineering from the University of Massachusetts Dartmouth, in 2015. He is currently pursuing the Ph.D. degree in computer science with Towson University. His research interests include big data, the Internet of Things, and security.

**CHENG QIAN** received the B.S. degree from Jianqiao University, Shanghai, China, in 2018. He is currently pursuing the M.S. degree with Towson University. His research interests include the Internet of Things, cyberspace security and privacy, and computer networks.

**WILLIAM GRANT HATCHER** received the B.Sc. degree in materials science and engineering from the University of Maryland and the master's degree in computer science from Towson University, in 2018, where he is currently pursuing the Ph.D. degree. His research interests include mobile computing and security, big data, and machine learning.

**WEI YU** received the B.S. degree in electrical engineering from the Nanjing University of Technology, Nanjing, China, in 1992, the M.S. degree in electrical engineering from Tongji University, Shanghai, China, in 1995, and the Ph.D. degree in computer engineering from Texas A&M University, in 2008. He is currently a Full Professor with the Department of Computer and Information Sciences, Towson University, MD, USA. Before joining Towson University, he was with Cisco Systems, Inc., for nine years. His research interests include cyberspace security and privacy, cyber-physical systems, the Internet of Things, and big data. He was a recipient of the 2014 NSF Faculty CAREER Award, the 2015 University System of Maryland (USM) Regents' Faculty Award for Excellence in Scholarship, Research, or Creative Activity, and the University System of Maryland (USM)'s Wilson H. Elkins Professorship Award, in 2016. His work has also received the Best Paper Awards from the IEEE ICC 2008, ICC 2013, IEEE IPCCC 2016, and WASA 2017.