IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received February 20, 2020, accepted March 3, 2020, date of publication March 6, 2020, date of current version March 18, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2978896

Convergence of Recommender Systems and Edge Computing: A Comprehensive Survey

CHUAN SUN[1], (Student Member, IEEE), HUI LI[1], XIUHUA LI[1,2], (Member, IEEE), JUNHAO WEN[1,2], (Member, IEEE), QINGYU XIONG[1,2], (Member, IEEE), AND WEI ZHOU[1]

[1]School of Big Data and Software Engineering, Chongqing University, Chongqing 401331, China

[2]Key Laboratory of Dependable Service Computing in Cyber Physical Society, Ministry of Education, Chongqing University, Chongqing 400044, China

Corresponding author: Xiuhua Li (lixiuhua1988@gmail.com)

This work was supported in part by the National NSFC under Grant 61902044 and Grant 61672117, in part by the National Key Research and Devlopment Program of China under Grant 2018YFF0214700, in part by the Chongqing Research Program of Basic Research and Frontier Technology under Grant cstc2019jcyj-msxmX0589, in part by the Key Research Program of Chongqing under Grant CSTC2017jcyjBX0025, and in part by the Major Science and Technology Program of Guangxi under Grant GKAA17129002.

ABSTRACT Under the explosive growth of information available on the Web, recommender systems have been used as an effective technology to filter useless information and attempt to recommend the most useful items. The proliferation of smart phones, smart wearable devices and other Internet of Thing (IoT) devices has gradually driven many novel emerging services which are latency-sensitive and computation-intensive with a higher quality-of-service. Under such circumstances, the data sources contain four key characteristics (i.e., sparsity, heterogeneity, mobility, volatility). The conventional recommender systems based on cloud computing are incapable of digging the information of user demands. Mobile edge computing is a novel computing paradigm via pushing computation/storage resource from the remote cloud servers to the network edge servers to provide more intelligent and personalized service. This paper comprehensively reviews the state of the art literature on the convergence of recommender systems and edge computing, and identify the future directions along this dimension. This paper can provide an array of new perspectives on the convergence for researchers, practitioners, and tap into the richness of this interdisciplinary research area.

INDEX TERMS Recommender systems, edge computing, the IoT, intelligent service.

I. INTRODUCTION

Recently, ubiquitous recommender systems are a vital and indispensable technology and application of Big Data and Artificial Intelligence (AI) [1]. In the explosive growth of data available on the Web, it can actively filter useless information and attempt to recommend the most useful items (e.g., news, products, services, etc.) to users via a recommendation algorithm [2]. A key feature of the recommendation algorithm is its ability to forecast a user's interests and demands based on analyzing user's personalized preferences, items features, user/items past interactions and other additional information [3]. Nowadays, the recommender systems start off becoming popular in various information access systems and has been successfully applied in pervasive across numerous web domains, such as e-commerce (e.g., Amazon, Netflix, Alibaba), information retrieving (e.g., Google, Yahoo, Baidu), social network (e.g., Facebook, Twitter, Weibo), location service (e.g., Foursquare, Yelp, Waze) and so on [4]. All in all, studies of the recommender systems have become a hot topic in academia and industry, forming a number of related research achievements.

In general, there are a set of *users* and a set of *items*, the recommendation algorithm computes a rank of each pair that indicates the correlation of a user to an item. The recommendation algorithm can be mainly categorized into collaborative filtering (CF), content-based recommendation (CB), knowledge-based recommendation (KB), and hybrid recommendation (HR) [5]. Each algorithm has advantages and limitations, for example, CF achieves the best accuracy of predictions about how much someone is going to enjoy a movie in *Netflix Prize*, however, it has sparseness and cold-start problems [6]. These algorithms usually collect all information of users and items into cloud servers to manage and analyze [7].

With the proliferation of smart phones, smart wearable devices and other Internet of Thing (IoT) devices, as well as the tremendous growth of Internet of Vehicles (IoV),

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

the number of IoT devices will reach nearly 50 billion on the Internet by 2020 [8], having gradually driven many novel emerging services such as virtual reality (VR), augmented reality (AR), live-stream, electric vehicle (EV), and smart city (SC). To promote the quality-of-experience (QoE) of users, the development of services is in intelligence, personalization, and integration directions. These services are latency-critical and computation-intensive with a higher quality-of-service (QoS). On the other hand, global mobile data traffic continues growth of nearly 11-fold over the next 5 years [9]. According to Cisco reports, nearly 850 *ZB* data will be generated each year by 2021, but the data center is only 20.6 *ZB* [10]. It indicates that the location distribution of data sources is undergoing a transformation from data centers to an expanding number of edge devices. Under such circumstance, the data sources contain the following key characteristics.

- **Sparsity**: in the edge environment, the historical data sources stored in edge server comes from a small amount or even one user's profiles. Many items' content missing causes that the data are not dense enough, but sparse enough [11].
- **Heterogeneity**: edge devices are produced by companies around the world. Each company has its systems, network protocols, and technical standards, resulting in the data sources are heterogeneity with high variability of data types and formats [12], [13].
- **Mobility**: the mobility is an inherent characteristic of users in the mobile networks. Smartphones and wearable devices attached to users will wander about, especially the sensors in IoVs move quickly. With the mobility of user, the data sources of user are distributed to different edge servers in different periods [14].
- **Volatility**: the state of the mobile edge network is volatility, when one user invokes a service many times, the QoS data may be different each time.

Although many existing recommender systems based on cloud computing have been proposed in traditional Internet environments, they are gradually unable to deal with these novel emerging services and massively distributed data in mobile edge network, they may fail to predict what users' interests and demands are. Due to the above challenging characteristics, the recommender systems are mainly facing the following three problems: 1) **cold-start problem**, as data sources of active users are usually very sparse, even new or inactive users lack relevant profiles, the cold-start problem occurs; 2) **exploration and exploitation problem**, for example, in online shopping, exploration implies recommending new goods and exploitation entails reusing existing goods. How to find an optimal trade-off between exploration and exploitation is crucial issue; 3) **security and privacy problem**, the data sources are produced by various IoT devices and distributed at different edge platforms, resulting in potential leakage of user data security problem. If all data sources are offloaded to a central cloud server, this centralized management may bring about privacy problem.

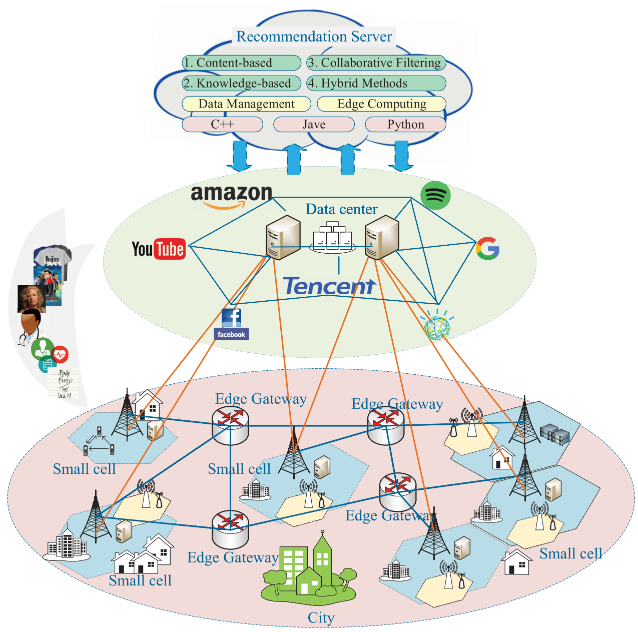

FIGURE 1. Recommender systems and edge computing.

Mobile edge computing (MEC) is a novel computing paradigm via pushing computation/storage resource from the remote cloud servers to the network edge servers to improve QoS. To process computation tasks and services as close as possible to data sources, edge servers can be deployed at or near base stations closer to users. MEC and conventional cloud computing are not mutually independent, but MEC is an extension and supplement of cloud computing [15]. Nowadays, various machine learning-based intelligent services have been deployed at edge servers to meet the critical requirements (e.g., agility, heterogeneous data analysis, and privacy-policy strategy) of computation tasks [16]–[18]. IoT-based devices and applications can understand user profiles deeply and comprehensively to personal demands of users, as the heterogeneous sources of users are available [19]. Meanwhile data sources processed at the edge servers not remote cloud servers potentially protects the privacy of users' information. Therefore, MEC is expected to solve the issues of current recommender systems. In fact, recommender systems are gradually being combined with edge computing as shown in Fig. 1.

Specifically, on one hand, recommender systems expect to deeply understand users' behaviors, demands, and interest via edge servers with the proximity of users [20], thus processing sparse, heterogeneity, mobility, and volatility data sources, solving cold-start, exploration and exploitation and security and privacy problem, providing perfect personalized services for users. There are many advantages as follows. 1) the devices of users can collect massive user data (i.e. health data, behavioral data, shopping data, etc.), due to the intimate relationship between users and devices, these user data truly reflect users' interests [21]; 2) edge computing distributed to the network edge can help to disseminate and collect user data and provide better data management. For QoS prediction in mobile network, edge servers can perceive the user mobility and make appropriate decisions to improve QoS prediction accuracy [22]; 3) the partial recommendation algorithm of recommender systems can be deployed at edge servers. Under this decentralized MEC, edge servers can perform data sources caching, aggregation, and lightweight processing to alleviate the pressure of the central cloud server [23]. Besides, recommendation results are responded by the nearest edge servers, eliminating the latency of data transmission between cloud servers and devices.

On the other hand, the recommender systems are key solutions towards overcoming information retrieval challenges. In edge computing, the edge servers and many third-party service providers are required to provide intelligent and personalized services for the unprecedented growth of IoT end-devices and data [24]. Recommender systems boost to more easily identify relevant association between users and items then finishing personalized services for meeting the functional and non-functional requirements [25], [26]. Edge caching a primary and indispensable technique in edge computing, which caches some popular contents at the edge servers in advance to reduce latency and network traffic, thereby improving the QoS [27], [28]. Under the limitation of caching capacity at edge servers, edge caching needs to optimize the cached content placement and content delivery, which primarily depends on the content popularity distribution and similarity of user demands [29]. Recommender systems are an efficient way to cache the most related and popular content at edge servers to maximize resource utilization and reduce latency by matching the requirements, hobbies, and habits according to users' environments [30].

Consequently, considering both of them face some same challenges and practical issues in multiple aspects, as illustrated in Fig. 2. We consider four enabling technologies for building recommender systems and Edge computing:

1) Recommender systems on Edge (recommender systems applications on edge), systematically organizing applications of recommender systems on edge.

2) Recommender systems in Edge (recommendation inference in edge), focusing on the deployment and inference of recommender systems in the edge computing paradigm.

3) Edge computing for recommender systems, utilizing decentralized edge computing architecture to meet higher requirements of recommender systems.

4) Recommender systems for Edge computing, which resorts to recommender systems for better edge caching by analyzing the relationship between the user and items.

With plenty of research investigating the convergence of recommender systems and edge computing, the current existing works urgently need to be summarized for further research. To the best of our knowledge, we are the first to make a comprehensive review that shapes the convergence of recommender systems and edge computing and position existing works and current progresses. The most related work is [21] and [31], however, they mainly focus on the recommender systems under IoT environment rarely involve edge computing. Different from [21], [31], this survey contributes on these respects: 1) we comprehensively review the state of the art literature on the convergence and investigate the holistic technical spectrum in terms of four enablers; 2) we summarize the deployment challenges of recommender systems by edge computing; 3) we discuss open issues currently limiting real-world implementations and identify the future directions along this dimension.

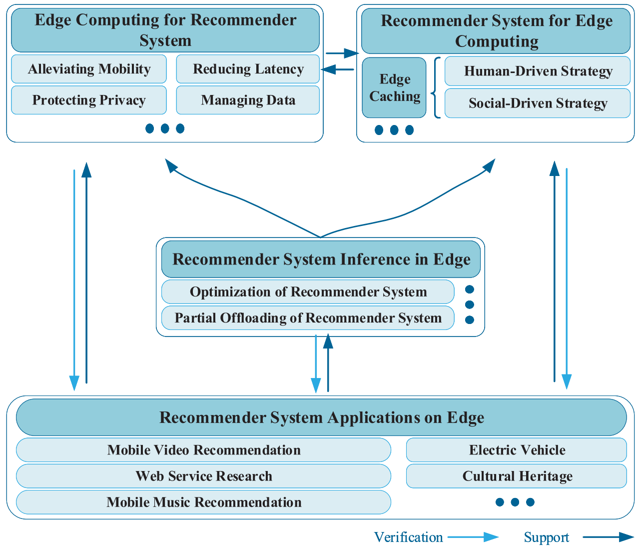

FIGURE 2. Conceptual relationships of recommender systems and edge computing.

The abbreviations in this paper are shown in Tab. 1. The remainder of this article is organized as follows. Section II and III respectively introduce fundamentals related to recommender systems and edge computing. We give four enabling technologies, i.e., recommender systems applications on edge in Section IV, recommendation inference in edge in Section V, edge computing for recommender systems in Section VI, and recommender systems for edge computing in Section VII. Section VIII collects and summarizes some datasets. Finally, we discuss future directions in Section IX and conclude the paper in Section X.

II. FUNDAMENTALS OF RECOMMENDER SYSTEMS

In the past 20 years, recommender systems have gained much attention. They are efficient software tools and techniques to overcome information overload and over-choice by proactively providing suitable suggestions for items to users. With the development of recommender systems, it has been successfully applied in various domains (e.g., e-government, e-resource, e-commerce, etc.) by multi-disciplinary efforts (i.e., artificial intelligence, human-computer interaction, data mining, etc.), shown in Fig. 3. Nowadays, more and more scholars participate in research on the related work of recommender systems. The number of studies on recommender systems has been increasing dramatically, depicted in Fig. 4.

TABLE 1. List of abbreviations in alphabetical order.

Abbr.	Definition	Abbr.	Definition
AI	Artificial Intelligence	NP	Non-deterministic Polynomial
CB	Content-based	QoE	Quality of Experience
CF	Collaborative Filtering	QoS	Quality of Service
ETSI	European Telecommunications Standards Institute	RPME	Replica Placement Strategy for Mobile Media in Edge Computing
EV	Electric Vehicle	SC	Smart City
HetRec	Heterogeneity and Fusion in Recommender Systems	SF	Smart Factory
HR	Hybrid Recommendation	SM	Smart Home
IoT	Internet of Thing	SVM	Support Vector Machine
IoV	Internet of Vehicles	SVD	Singular Value Decomposition
IT	Internet Technology	DNN	Deep Neural Networks
KB	Knowledge-based	YFCC100M	Yahoo Flickr Creative Commons 100 Million
MEC	Mobile Edge Computing	ZB	Zettabytes

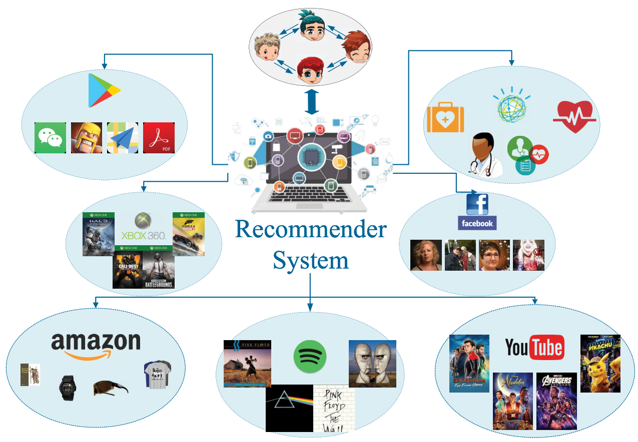

FIGURE 3. The most common recommender systems applications.

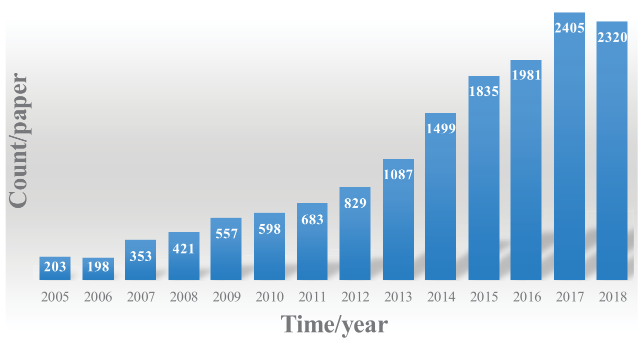

FIGURE 4. Popularity of recommender systems in recent years. We search in "*Web of Science*" with "*recommender systems*" as keywords. The search results show the number distribution chart of papers from 2005 to 2018.

A. WHAT ARE RECOMMENDER SYSTEMS?

There are various definitions and explanations of recommender systems. For example, according to Wikipedia, recommender systems are a subclass of information filtering system that seeks to predict the "rating" or "preference" a user would give to an item. However, the definition is somewhat obscure and not exactly right. It defines "*What is it?*" with "*How does it work?*", and this stipulates the way to implement the recommender systems in disguised form.

We can redefine recommender systems with the following three questions from another perspective.

1) What can it do? Recommender systems can proactively find out in advance the connection that will eventually arise between the users and the items. The connection is not strictly delimited to a certain relationship, instead, any relationship can be considered the connection [32]. Because myriads of things are connected. People tend to make more connections, spawning numerous social applications; consumption connections between users and goods become more common in e-commerce, making a number of unique e-commerce applications.

2) What does it need? Recommender systems exploit existing connections to explore and predict future connections. An information content application (TouTiao.org), for instance, expects users to click on the massive content pages. Each click and reading are connections. Connections of different levels and importance are constantly built with recommender systems. Then, it provides accurate and personalized suggestions for news to be of reading to a user.

3) How does it work? The definition of Wiki provides one way by predicting users' ratings and interesting. They are two main components of recommender systems. Recommendation algorithms are usually classified into four categories: content-based, collaborative filtering, knowledge-based, and hybrid methods.

B. RECOMMENDATION TECHNIQUES

1) CONTENT-BASED RECOMMENDATION

CB recommendation techniques learn to recommend items that are similar to items previously preferred by a user. For example, the music recommender system analyzes the information of music (i.e., textual and audio) in a user's playlist, then recommends similar genre musics according to some content similarity measures [33]. All CB recommendation techniques follow the basic principles: **a**. As much as possible users' data resources acquisition; **b**. Data resources cleaning; **c**. Data resources mining; **d**. Similarity calculation. To calculate the degree of similarity, there are traditional information

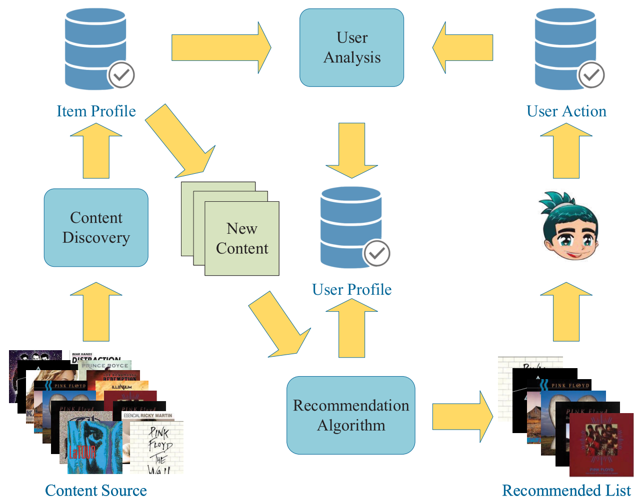

FIGURE 5. The framework of a content-based recommender systems.

retrieval methods, statistical learning and machine learning methods. The main advantages are user independence, transparency, and new unrated item recommendation, but suffer new-item problems [34]. A typical CB recommender system framework is shown in Fig. 5.

2) COLLABORATIVE FILTERING

CF recommendation techniques are based on the idea that analogous users have analogous demands and interests [35]. These techniques recommend to the active user items that other user shares similar demands. The similarity of two users is calculated based on the similarity in users/items historical interactions, either explicit (e.g., users' previous ratings) or implicit feedback (e.g., browsing history). CF can be divided into two categories: memory-based models and model-based models. The memory-based models can remember all users' profiles and recommend similar items to them, or recommend items that similar users like to them [36]. The model-based models learn a generalized model from the Matrix of users and items to fill the blanks, shown in Fig. 6. CF achieves the best accuracy of predictions about how much someone is going to enjoy a movie in *Netflix Prize*, however, it has sparseness and cold-start problems [6].

3) KNOWLEDGE-BASED RECOMMENDATION

Knowledge-based recommendation techniques offer items to users based not on users' ratings history, but specific knowledge about the users, items and their relationships [37]. It might prompt the user to give a series of rules or guidelines on what the results should look like or an example of an item. The system then searches through its database of items and returns similar results. If you look for a house, for instance, when you input price, area, location, or how many bedrooms, etc., the website returns a list of houses based on those constraints. Hence, these techniques do not suffer ramp-up/cold-start problem [38].

FIGURE 6. The matrix of different users and items. Each row represent a user, and each row represent every item in the catalogue.

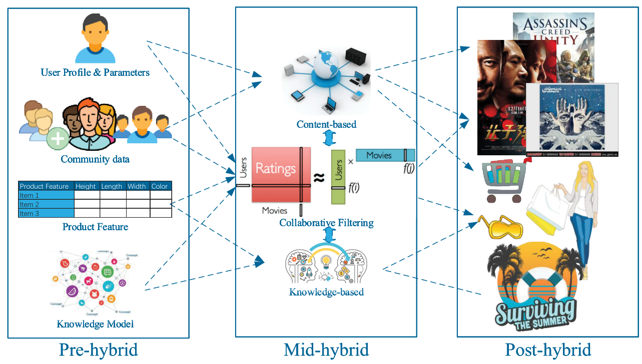

FIGURE 7. Relationships between different hybrid methods.

4) HYBRID RECOMMENDATION

HR methods hybridize the features of two or more recommendation techniques to overcome the drawbacks of traditional recommendation techniques and achieve better performance [39]. The common HR methods are divided into three strategies: post-hybrid, mid-hybrid, and pre-hybrid [40]. The post-hybrid combines recommendation results generated by two or more recommendation techniques to output suggestions based on some combination ways. It is the combination of the decision level. The mid-hybrid is the combination of two or more recommendation algorithms at the model level. For example, the hybrid method combines CB recommendation algorithm based on CF to mitigate the sparseness problem. The pre-hybrid directly combines many recommendation techniques into a uniform model, which is a feature-level combination. The hybrid methods can be illustrated as in Fig. 7.

Recently, many advanced recommendation techniques have been developed to solve more complex problems, such as fuzzy-based, context aware-based, trust aware-based, social-network, and deep learning-based, etc. The evolution of recommender systems is shown as Fig. 8. The advantages and disadvantages of different recommendation techniques are shown in Tab. 2.

TABLE 2. Comparison of different recommendation techniques.

Techniques	Type of knowledge	Advantages	Drawbacks
Content-based [33], [34]	User profile	Dependent, transparent, new recommendation	Overspecialized, limited content analysis
Collaborative Filtering [35], [36]	Similarity matrix	Easy to create, diverse, no knowledge required	Sparseness, scalability and cold-start
Knowledge-based [37], [38]	Decision rules	Simple and direct	Bottleneck in extracting knowledge
Hybrid methods [39], [40]	Above all	Robust, improve performance	Complexity, difficult to manage the system

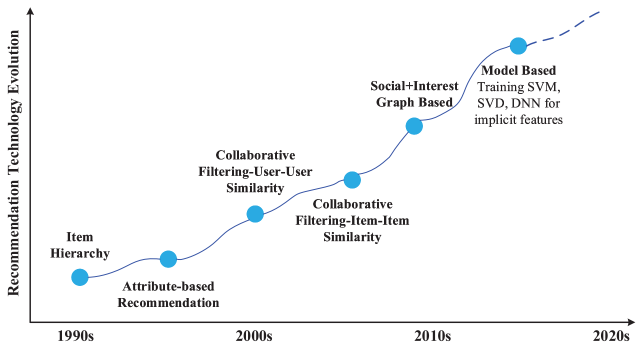

FIGURE 8. Evolution of recommender systems.

C. RECOMMENDER SYSTEMS IN IoT

In the past, the fast growth of the Internet is mainly based on centralized cloud architecture. The centralized cloud servers are equipped with powerful computing resources and massive storage capacity. Meanwhile, a lot of Internet Technology (IT) companies (e.g., Google, Microsoft, IBM, Baidu, etc.) provide excellent cloud services, resulting in most computation, storage, and control done in the cloud servers [41]. Cloud computing came into being due to investment from industry and academic. From the cloud provider's perspective, cloud computing has advantages such as resource consolidation, uniform management, and cost-effective operation [42]. Hence, the most conventional recommender systems are based on cloud computing, not only recommendation techniques researches but also systems applications [43]. For instance, Amazon the world's largest online marketplace collects and stores the users browsing and purchase profiles at the central cloud servers, and the company investigates a powerful recommender system by analyzing all users' profiles to recommend user interested items.

As various novel emerging services are usually latency-critical and computation-intensive in the IoT era, the data continuously generated on a very large-scale from various sources has four key characteristics, including sparsity, heterogeneity, mobility, and volatility. On the one hand, the conventional recommender systems processing services one by one based on cloud computing are gradually unable to deal with these emerging services. On the other hand, recommender systems can provide effective and efficient data mining and decision-making for massive IoT resources and services [31]. Due to the data challenging characteristics,

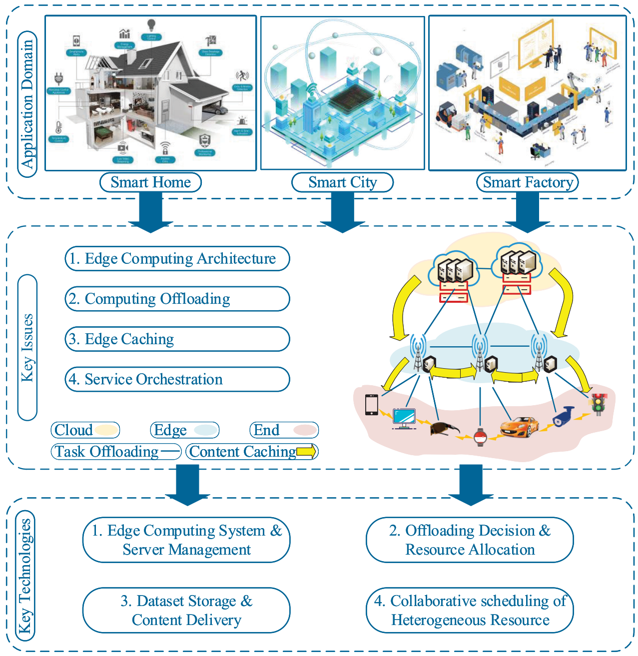

FIGURE 9. Edge computing framework.

the recommender systems are mainly facing three problems: cold-start, exploration and exploitation, and security and privacy. The two are facing challenges to each other. If these problems can be solved, the role of the recommender systems in IoT will be fully utilized. It is desirable to investigate the novel recommender systems paradigm to proactive discovering and suggestion.

III. FUNDAMENTALS OF EDGE COMPUTING

Edge computing is a novel computing paradigm via pushing computation, storage, and network resources from the remote cloud servers to the network edge servers [44]. Edge computing is a promising technology to mitigate bottlenecks of conventional cloud computing, such as backbone network congestion, traffic heavy load, high transmission latency, etc., by releasing the cloud servers pressure to the edge servers to reduce latency and improve the QoS [45]. Edge computing has been significantly developed and researched in both industry and academia. This section will briefly introduce edge computing from the following three aspects: application domains of edge computing, key issues, and key technologies. The framework is shown as Fig. 9.

A. APPLICATION DOMAINS OF EDGE COMPUTING

1) SMART HOME

The smart home (SM) is an ideal lifestyle which can access all home status via many IoT devices (e.g., temperature sensors, security system, lighting system, etc.) and automatically control home environment to improve security, convenience, and comfort [46]. Recently, more and more companies have launched a wide range of SM devices that are heterogeneous. Meanwhile, many of SM devices have serious security issues [47]. Edge computing becomes a suitable network platform for smart home, due to meet the requirement that enables applications to be processed locally through the network [48]. Cao *et al.* [49] propose an edge operating system for SM and further discuss challenges.

2) SMART CITY

In 2017, Alphabet builds a high-tech zone named Quayside and devotes to make as a SC model [50]. Edge computing has been significantly developed in a smart city to mitigate the bottlenecks of data processing, such as massive, mobile, heterogeneous. Edge computing can intelligently control traffic system to ease traffic jams. In the self-driving car, edge computing can provide real-time vehicle control [51]. The study in [52] proposes a hands-on demo SC application to solve people's movement.

3) SMART FACTORY

As the concept of the smart factory (SF) comes from the integration of conventional industrial and Internet. It has higher requirements consisting of low latency, high reliability, high privacy, and massive connectivity [53]. The Industrial Internet Consortium publishes a white-book and points out edge computing is a promising solution for SF [54]. Lots of computation can be executed at the network edge to shorten delay of industrial applications. Moreover, edge computing can promote industrial production methods progress [55].

B. KEY ISSUES AND KEY TECHNOLOGIES

Significant research and development efforts in edge computing aim to improve network efficiency. However, edge computing has not yet reached a consensus on standardized architectures, definitions, and protocols. There are still many key issues should be studied.

1) EDGE COMPUTING ARCHITECTURE

Edge computing can provide software-defined applications and cloud computing services at the network edge. Edge computing aims to become the infrastructure with computation/storage resources to meet various requirements of most users, instead of individual solution [56]. Especially, edge computing provides a programmable ecosystem via modular and open solutions to improve QoS. And service providers are allowed to collect more users' information. The study in [57] introduces the edge computing architecture that can be classified into Small Cell Cloud, Mobile Micro Clouds, Fast Moving Personal Cloud, Follow Me Cloud, CONCERT, and ETSI MEC.

2) COMPUTATION OFFLOADING

Computation-intensive tasks execute at resource-constrained end-devices, which will consume huge resources and energy. To mitigate this issue, some tasks can be offloaded from the end-devices to the network servers to reduce the end-devices resource consumption and latency [58]. But it is not easily realized even by edge computing [16]. The main bottleneck lies in that the task offloading is limited by computing, communication and caching capacities of the edge servers as compared with cloud computing, and raises the optimization problem of server selection and resource allocation. Many academics propose a set of offloading algorithms to optimize delay or resource. Zheng *et al.* [59] formulate computation offloading process as a stochastic game and propose a multi-agent stochastic learning algorithm to achieve optima offloading decision. Deep reinforcement learning is the most successful application in computation offloading [60].

3) EDGE CACHING

Current content delivery technologies can optimize content transmission service to improve the server's availability and reduce latency, but they can not make quick adjustments based on users' status. Edge caching dynamically optimizes content delivery service based on the network status and wireless channel load [61]. From the user's perspective, since the edge servers are deployed at the network edge and are very close to end-devices, it can combine user mobility and content access logs to optimize the behavior. There are many existing works investigating the issues of edge caching. Li *et al.* [62] propose the concept of ''Caching-as-a-Service'' which can cache anything at anytime, anywhere in the cloud-based 5G mobile systems. Wang *et al.* [63] investigate the issue of the edge caching optimization of hierarchical wireless networks. They propose a Q-learning based distributed content replacement strategy to satisfy the large-scale real trace.

4) SERVICE ORCHESTRATION

The edge computing platform is deployed in the mobile network, resulting in many issues about service orchestration. Due to the increase of edge nodes in the network and user mobility, the system becomes more complex to efficiently manage each edge server resource (computation/storage). Meanwhile, edge computing should equip with life-cycle management, in other words, authorize instantiation of third-party applications or terminate application service requests on-demand [64]. With the goal of user experience quality and service reliability, it is essential to orchestrate the services of the edge computing platform in combination with resource management and service deployment solutions.

IV. RECOMMENDER SYSTEMS APPLICATIONS ON EDGE

In general, recommender systems are mainly based on cloud computing for meeting user's demands and interests, since the mature cloud servers provide powerful resources (e.g., computation, storage, communication, and user data sources) for complex and heavy computing recommender systems. However, conventional recommender systems are gradually unable to meet the requirements of IoT services. Recently, a novel computing paradigm has been proposed by pushing computation and storage resources from the central cloud servers to network edges. Hence, deploying recommender systems applications at the edge servers can perform some lightweight processing to improve QoS. We present some applications related to recommender systems and edge computing as the following.

A. MOBILE VIDEO RECOMMENDATION

Video recommendations can greatly enrich our cultural life, recommending favorite movies to enjoy life, suggesting online-courses to learn, and forwarding interesting videos to relax. However, more and more users are used to watching videos on mobile devices, not computers, resulting in hardly discovering users' demands. Some researchers focus on how to develop video recommendations for mobile devices [65]. RPME [66] is proposed to relieve the pressure of cloud computing platforms, such as massive equipment requirements for access, massive data, insufficient bandwidth, and high power consumption. RPME combines edge computing for dynamically placing the replicas to improve QoE of users. Yang *et al.* [67] investigate a movie recommender system that can protect the security of movie/video content during transmission. To meet the requirements of movie/video content transmission delay, they adopt a distributed parallel processing model based on edge computing for movie applications and services. Meanwhile, they perform the movie adaptation/encoding transforms in edge servers to recommend for heterogeneous mobile devices. In [68], the study proposes an edge computing intelligent cache by jointing an innovative composite recommendation algorithm to streaming media recommendation. The recommendation algorithm can decide which media files should be cached at the caching-limited edge servers for users to reduce latency and improve QoS.

B. ELECTRIC VEHICLE

The EV has been significantly developing due to comfort, safety, green, and alleviating traffic congestion. Both industry and academia are interested in the development of EV for the SC [69]. There are several challenges to realize the intelligent EV, such as charging service, intelligent navigator, and so on. The sustainable charging of EV is crucial and current charging infrastructures is underutilized, the study in [23] investigates an intelligent EV charging recommendation based on edge computing supporting architecture to enhance the utilization of charging infrastructures. This charging recommendation combines edge computing and the cloud computing to optimize complex computation. The edge servers deployed in road side units are designed as a decentralized edge computing-based information communication technology framework to perform information caching, aggregation, and lightweight processing. A global controller based on cloud computing can facilitate the prediction accuracy of the charging availability of charging stations. Intelligent navigator is a key component in electric vehicle for suitable human-computer interaction. In [70], a weighted interest degree recommendation algorithm is proposed by using association rules to improve the accuracy of recommendation content. Moreover, the recommender systems can reduce the time of search and screening the massive retrieval of information.

C. WEB SERVICE RESEARCH

Web service is a basic unit of service computing deployed in the complex Internet environment. It is a new application model directly called for decentralized computing and an effective mechanism for the data and service integration on the web. Hence, it is important and necessary to research the new architecture of web services, on the combinations with other good techniques, and the integration of services [71]. To improve accuracy expression of user preference, the study in [72] proposes a cultural distance-aware service recommendation algorithm utilizing joint factors of similarity and the local characteristics of users. They consider that users may be similar behavior in the same edge server and different users in other edge servers. So a cultural distance approach expresses user preference based on edge computing. In [73], Zhou *et al.* investigate an intelligent service recommendation method based on CF in the edge computing environment. To solve the lack of feedback in the edge environment, they propose an inverse CF to smooth the cold-start recommendation process.

D. OTHER APPLICATIONS

Current applications about recommender systems on edge computing are open domains and involve some interesting researches. Su *et al.* [12] investigate an edge intelligence empowered recommender system for enabling cultural heritage applications. This system can recommend the tourism items of cultural heritage, such as specific ruins of an archaeological site, sculptures and/or pictures exhibited within a museum, historical buildings, and famous squares in downtown, when you travel all over the world. In [27], the study applies a combination of recommender system and edge computing to uniquely yours of music. The online social networks application is purposed based on a collaborative user-centered framework [74].

V. RECOMMENDATION INFERENCE IN EDGE

To further improve the recommendation accuracy, recommender systems become complex and require larger-scale users' profiles. The conventional recommender systems-based cloud computing with dramatic computation costs can not cope with the case of IoT. They can not be deployed in the edge with limited resources. Meanwhile, there are some new problems to decrease the accuracy, such as the potential leakage of user privacy and mobility of user data. Therefore, recommender systems should be designed sophisticatedly and further customized to fit in the resource-constrained edge to meet these new problems.

A. OPTIMIZATION OF RECOMMENDER SYSTEMS

In [72], Wu *et al.* introduce a cultural distance-aware service recommendation approach to improve accuracy expression of user preference. The conventional CF only focuses on the similarity of the rating matrix, which is not suitable for services in the edge network environment. They jointly utilize the similarity, local characteristics and behavior of users to predict the rating matrix and recommend the services with higher ration. To overcome the sparsity of the edge server, the approach also introduces a missing rating prediction method based on CF. In [75], the study investigates a novel recommender system based on mobile App data sources for personalized offers. The system is designed as three steps, including in data gathering from mobile devices, building a digital inventory, and creating personalized recommendations. In creating personalized recommendations, a hybrid method (collaborative and content-based) is proposed to compute a list of personalized recommendations and feedback to the device by learning user demands and behavior. However, this system does not take the privacy problem into account.

Due to the service usage data collected from various mobile devices and distributed edge platforms, the potential leakage of user privacy may arise during the recommender systems process. In [76], Gong *et al.* propose an improved locality-sensitive hashing-based service recommendation method to protect users; privacy over multiple quality dimensions during the distributed mobile network. In [77], Jiang *et al.* propose a secure friend recommender system without exposing the actual user behavior but the anonymous data to protect privacy data. The anonymous data is from Chinese ISP recording the user browsing behavior.

In [20], Yao *et al.* introduce the new properties of the recommender system in IoT and propose a unified probabilistic factor-based framework. By integrating the relationships between heterogeneous entities of IoT, a hypergraph was designed to model things' spatiotemporal correlations. The spatiotemporal association can generate implicit object associations, establish an IoT experimental platform, and verify the possibility and efficacy of the method. A large amount of IoT data from polyphase data initiators were converged and transmitted to online or offline systems, which resulted in increased intricacy of data storage and query, which was especially suitable for spatiotemporal data handling via online or offline systems.

To further improve QoS of edge computing with changing attributes, the study in [78] proposes a context-aware multi-QoS prediction method combining the QoS attributes and three contextual factors (i.e., task type, task volume, service workload) for services computing in edge computing. The service workload is predicted by an optimized support vector machine.

B. PARTIAL OFFLOADING OF RECOMMENDER SYSTEMS

Matrix factorization is an efficient approach for CF due to good performance on processing sparse matrices and prediction accuracy. But its computation complexity and model training take lots of time and resources, so it only is deployed at the cloud server, which results in many issues about user privacy and updating the model with new QoS information. Meanwhile, there are many other problems that the algorithm is deployed at the cloud server rather than at the edge server, which leads to more failures in a dynamic service network. To meet these problems, the study in [79] investigates a stacked autoencoder with dropout on a deep edge computing framework. Comparing conventional matrix factorization methods, the proposed method is a dramatic reduction in training time. Hence, it can be deployed at the edge server and maintain good performance in service compositions.

VI. EDGE COMPUTING FOR RECOMMENDER SYSTEMS

Extensive deployment of recommender systems, especially mobile network services recommendation, require shorter response time and higher performance. These systems need the support of edge computing, including the data management level, the control level, the system architecture level, optimization of edge software. The current research mainly focuses on the following four directions: 1) Alleviating mobility; 2) Protecting Privacy; 3) Reducing latency; 4) Managing Data.

A. ALLEVIATING MOBILITY

In the mobile network, the mobility is the inherent attribute of users, which decreases the accuracy of QoS prediction. In [14], Wang *et al.* propose a service recommendation method based on CF for QoS prediction. They utilize the fact that users invoke different edge services in dynamically changing locations. The edge computing can transmit the user profile between different edge servers to solve user mobility problem. They design a hybrid dataset of the Shanghai Telecom and WSDream to prove the effectiveness of the proposed method. In [27], the study proposes content-oriented caching based on a recommendation approach for wireless communications. To solve user mobility and the randomness of contact duration, they design an AI-based approach to predict where content is cached.

B. PROTECTING PRIVACY

Many companies provide different cloud services based on own cloud platforms, however, they are reluctant to share the original data due to privacy concerns. Privacy protection is fundamental and paramount for a good recommender system. In [80], Zhou *et al.* propose a tree-based privacy-preserving and trustworthy distributed multimedia contents retrieval system. In [81], Qi *et al.* propose a distributed locality-sensitive hashing-based method for service recommendation. The distributed user data is processed by distributed locality-sensitive hashing in different edge servers to protect privacy. In [82], the study investigates a novel time-aware and privacy-preserving service recommendation approach based on LSH.

C. REDUCING LATENCY

Service latency is the main factor that affects QoS and QoE of the user. Edge computing is an effective paradigm to guarantee the high efficiency requirement of transmission delay. In [67], Yang *et al.* design a multimedia recommendation and transmission system. To improve QoS of multimedia service, they design an edge computing framework between the cloud data center and user devices to store and transmit multimedia data. Due to the edge servers closer to user, it significantly achieves a shorter latency. In [83], Wu *et al.* investigate an online media recommendation solution applied in real-life. Their system is significantly improved in training and execution by pushing partial machine learning procedures to the edge-cloud testbed.

D. MANAGING DATA

In the era of IoT, the data sources are complexity and variety with the main four features (sparsity, heterogeneity, mobility, volatility). The conventional recommender systems based on cloud computing can not efficiently analyze the massive data. To cope with collection and analysis of distributed and heterogeneous data sources, Su *et al.* [12] propose an advanced data management technique based on an edge intelligence paradigm. They exploit jointly recommendation techniques and edge artificial intelligence facilities to improve data management and accuracy. In [84], the study proposes a healthcare management system to recommend healthcare services, which can process massive amounts of data using model transformations.

VII. RECOMMENDER SYSTEMS FOR EDGE COMPUTING

More and more users are used to acquiring service content through mobile devices. The proliferation of users and contents makes the backbone network congestion of conventional cloud computing. Edge computing is a promising paradigm to reduce the transmission delay in the era of IoT, where contents are cached at edge servers close to users. However, it is not easy to realize edge caching under massive content analytics. 1) What contents should be cached? It is important but difficult to choose contents cached with the constraint of the capacity of the edge server cache. 2) How to make a caching strategy? Due to the data volume of contents cached at the edge server, it becomes complex to recommend user-preferred contents and clear useless contents. The recommender systems are key solutions to overcoming information retrieval challenges. It can dramatically support edge caching in edge computing. However, there is a phenomenal conflict that the goal of edge computing and recommender system is not the same. The edge caching seeks to maximize all users' demands, but the recommender systems focus on individual user's demand. There are two optimization strategies: human-driven and social-driven.

A. HUMAN-DRIVEN STRATEGY

In [85], the study investigates a userspace customized recommendation service platform. To directly improve QoE of user, the edge caching integrates the recommender systems which can select the user preferred items to reduce service request time. However, the study only proposes a platform, without theoretical research and experimental verification. In [86], Pu *et al.* advocate content retrieval at the edge servers to overcome the problem of explosive growth of mobile data traffic. They propose a social-aware named data framework with Friendship Circle by searching high content similarity to improve delivery ration and reduce average delay.

In [87], Bai *et al.* propose a novel hyper-graph framework to enhance the benefit of lowering energy consumption and improving QoE of users. The framework takes social ties among users and common interests into consideration for devices caching based D2D communication scheme. The interest similarity impact includes multi-dimensional and user's demand, a clustering and recommender system functionality is designed to mitigate the impact and enrich the accuracy of user mobility analysis.

In [66], Tang *et al.* investigate an effective replica placement mechanism for mobile media streaming in edge computing with the constraints of the capacity of edge servers caching and the service capacity of the requested replica. The optimization goal of this mechanism is to improve recommendation accuracy. They utilize the user information and the user-item rating matrix to cluster the users, then generate a replica recommendation sequence. This mechanism performs well in load balancing. reducing the average response delay, and improving QoS.

B. SOCIAL-DRIVEN STRATEGY

In [88], the study early studies the impact of vehicular traffic demand on 5G caching architectures. They aim at maximizing the cache size and research that network topology and caching architecture have the largest impact on the total cache size, including in individual base stations, base station rings, aggregation-layer pods, and core-layer switches. However, they do not deeply study the recommender system in edge caching. The user demand information is given by track of the popularity of each content item within each cell.

In [92], Liu *et al.* investigate the content caching and recommendation at base stations to maximize the caching gain based on the successful offloading probability. To guarantee QoS and QoE of users, a model is proposed to capture the impact of user' interests. A hierarchical iterative algorithm is proposed to solve the integer programming problem with fixed-threshold. To push content from base stations to users, they investigate the proactive caching at mobile devices to recommend personalized content to user [93]. They optimize jointly content pushing and recommendation to maximize the net profit of a mobile network operator. They propose a

TABLE 3. Related datasets supporting further research of the recommender systems and edge computing.

Dataset Name	Link	Description	Reference
SNAP	http://snap.stanford.edu/data/index.html	It collects more than 50 large network datasets from thousands of nodes and edges. Besides, it provides powerful open source tools. It can be used for the research of complex networks. However, the C++ programming of the tool poses challenges for researchers who usually use Python.	[22]
YFCC100M	http://projects.dfki.uni-kl.de/yfcc100m/	This dataset collected by Yahoo Flickr includes 100 Million about the computer vision and multimedia research. It's worth noting that the dataset is all composed of text data, not pictures or videos. The dataset has geographical information, which can provide the research on edge caching based on edge nodes at different locations.	[80]
WeFi	http://www.wefi.com	WeFi provides its users with information on the safest and fastest Wi-Fi access points available at the user's location. This dataset contains exact location, wi-fi name and category for each Wi-Fi network, as well rich QoS information such as download speeds, latencies and more. It are very suitable for the research on the convergence of recommender systems and edge computing.	[88]
YouTube	http://netsg.cs.sfu.ca/youtubedata/	The dataset contains several information about the videos of YouTube. It is widely used in the research of recommendation systems. How to combine edge networks is open issue.	[89]
HetRec	https://grouplens.org/datasets/hetrec-2011/	These datasets contain social networking, tagging, and resource consuming information from sets of around 2,000 users.	[86]
GeoLife GPS Trajectories	https://www.microsoft.com/en-us/download/details.aspx?id=52367	This GPS trajectory dataset was collected in Geolife project by 182 users in a period of over five years (2007-2012).	[90]
WS-DREAM	http://wsdream.github.io/	WS-DREAM contains 3 main components: QoS prediciton, log management, and review mining. It is very famous recommendation systems researchers. Hence, based on these existing works, it can be easily used in various researches on edge computing.	[13], [76], [81]
MovieLens	https://grouplens.org/datasets/movielens/	MovieLens is a dataset about movie ratings, which contains user ratings information on movies obtained from IMDB. It is as widely used in recommender systems as WS-DREAM dataset. The dataset is easy to use due to the clear structure.	[11], [66], [74], [89], [91]

reinforcement learning framework to learn user performance. However, solving the joint problem with reinforcement learning will cause the curse of dimensionality. To circumvent the problem, two reinforcement learning models are designed.

There is a small population of user and the local population of edge servers in content edge caching. Two questions need to be considered: 1) how to determine the local population of edge server by the small population of user? 2) how to accurately predict the local population of the next period? In [94], Guo *et al.* investigate an edge caching with a recommendation at the base station, and the optimization problem is to maximize cache efficiency whereas not violating user preference. They propose a temporal-spatial recommendation policy with deep reinforcement learning to guide users to access the preferred content in proper time and place.

Due to the user mobility, the current edge server serving a user may not store the requested content, but nearby edge servers can meet user's requirement. Hence, the study in [89] investigates collaborative edge caching with recommender system to improve the performance of edge servers. They propose the concept of soft cache hits to satisfy the user instead satisfy every possible user request. A generic model for mobile edge caching with soft cache hits is proposed to capture more substitution devices. Meanwhile, they prove the optimal problem of edge caching is NP-hard, and efficient approximation algorithms solve this problem. The study in [91] proposes a simpler heuristic algorithm to solve the NP-hard problem.

VIII. DATASET ANALYSIS

There are many public datasets that can be used for interdisciplinary research of recommender systems and edge computing. To facilitate further research, we collect and summarize some datasets for those researchers, practitioners, and tap into the richness of this interdisciplinary research area, shown in Tab. 3. The Tab. 3 provides detail information on some commonly used datasets, consisting of name, link, description, and reference. We provide some short comments such as advantages and disadvantages, and these notions can help novices quickly determine which dataset is suitable for further research. Most datasets contain relationship between users and contents and are used for edge caching, e.g., YFCC100M [80], YouTube [89], WS-DREAM [76], etc. Meanwhile, some datasets can be used for recommender systems research in edge computing environment. Researchers can mix design and use these datasets according to their research problems. In summary, how to design the dataset for research is still an open issue.

IX. FUTURE DIRECTIONS

In retrospect of the convergence of recommender systems and edge computing, we discuss issues related to future research to identify unsolved issues and envision several promising directions.

A. MORE PROMISING APPLICATIONS

Compared to the conventional recommender systems being applied to various domains, the recommender system based on edge computing is emerging. The applications mainly focus on a small part of the smart city, such as the mobile phone and the electric vehicle. There are many promising applications not dabbled yet, such as the smart home and the smart factory. The smart home contains a lot of recommendations and is worth being deeply studied. In the smart home, various sensors can collect comprehensive information about user and environment. The information is analyzed at nearby edge nodes to protect user privacy and deeply mine user's behavior. All smart IoT devices are controlled by the recommender systems to generate comfortable living conditions. Besides, the research on the recommendation of the smart city should be deepened to improve convenience. In the smart factory, manufacturing systems usually have many problems, such as excessive item retrieval, redundant information, and time-consuming manual screening [95]. The combination of the recommender system and edge computing is a promising solution to meet manufacturing requirements and reduce delay.

B. DISTRIBUTED RECOMMENDER SYSTEMS

With the development of communication technology, more and more data are collected from distributed devices and require considerable efforts to be aggregated for centralized analysis [96]. Many studies have investigated the optimization of the recommender systems in the mobile edge network. The trend is to deploy the recommendation algorithms run on edge servers or end-devices (e.g., smartphones and smart IoT devices) to provide context-aware services other than a centralized server. Therefore, how to offload the partial distributed recommender system to different edge servers is an open issue. The entire system runs on multiple edge servers instead of a central server. Under this circumstance, the recommender system needs to simultaneously support massive edge devices. How to improve the scalability of the system is a question worth considering. Meanwhile, the hardware and software architectures should also be developed to support the novel distributed recommender system deployment.

C. ADVANCED EDGE COMPUTING

Deep learning has been extensively used for various tasks including items recommendation in recommender systems [4] and computation offloading in edge computing [97]. However, there are few studies on the convergence of two domains based on deep learning. Deep learning emerges and draws increasing attention on various occasions for three main reasons. As the data become more complex, there is a pressing need to analyze data that cannot be easily understood by humans. It could also be impossible for the human to manually extract and define the features to train a model. Another aspect that relationships between humans and devices are getting closer and fuzzy. How to bridge this gap between end-users and devices is a key problem. The last reason is that edge computing has two bottlenecks about task offload and edge caching. And two bottlenecks become more and more complex to achieve an optimal strategy. Hence, the advanced edge computing based on deep learning for the recommender system should be investigated to meet the above challenges [98].

X. CONCLUSION

Edge computing and recommender systems, as promising solutions for the era of big data, are expected to support each other and improve the QoS. Both recommender systems and edge computing are ongoing hot research topics in recent years. In this paper, we provided a comprehensive review of the most notable works about recommender systems in the mobile edge network. We highlighted the unique challenges of the recommender systems brought by the edge environment and proposed a classification scheme for organizing and clustering existing publications. Additionally, we discussed issues related to future research to identify unsolved issues and envisioned several promising directions. To facilitate further research, we summarized the information of related datasets. As the edge computing becomes common and powerful, the recommender systems based on edge computing will play an important role. The goal of this survey is to provide readers a quick grasp of state-of-the-art and promising research to cause widespread concerns and discussions.

REFERENCES

[1] L. Ogiela and M. R. Ogiela, "Data mining and semantic inference in cognitive systems," in *Proc. Int. Conf. Intell. Netw. Collaborative Syst.*, Sep. 2014, pp. 257–261.

[2] J. Bobadilla, F. Ortega, A. Hernando, and A. Gutiérrez, "Recommender systems survey," *Knowl.-Based Syst.*, vol. 46, pp. 109–132, Jul. 2013, doi: 10.1016/j.knosys.2013.03.012.

[3] J. Lu, D. Wu, M. Mao, W. Wang, and G. Zhang, "Recommender system application developments: A survey," *Decis. Support Syst.*, vol. 74, pp. 12–32, Jun. 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167923615000627

[4] S. Zhang, L. Yao, A. Sun, and Y. Tay, "Deep learning based recommender system: A survey and new perspectives," *ACM Comput. Surv.*, vol. 52, no. 1, pp. 5:1–5:38, Feb. 2019, doi: 10.1145/3285029.

[5] J. K. Tarus, Z. Niu, and G. Mustafa, "Knowledge-based recommendation: A review of ontology-based recommender systems for e-learning," *Artif. Intell. Rev.*, vol. 50, no. 1, pp. 21–48, Jun. 2018, doi: 10.1007/s10462-017-9539-5.
[6] A. Said, B. J. Jain, and S. Albayrak, "Analyzing weighting schemes in collaborative filtering: Cold start, post cold start and power users," in *Proc. 27th Annu. ACM Symp. Appl. Comput. (SAC)*. New York, NY, USA: ACM, Mar. 2012, pp. 2035–2040, doi: 10.1145/2245276.2232114.
[7] X. Zheng, L. D. Xu, and S. Chai, "QoS recommendation in cloud services," *IEEE Access*, vol. 5, pp. 5171–5177, Apr. 2017.
[8] "Fog computing and the Internet of Things: Extend the cloud to where the things are," Cisco, San Jose, CA, USA, White Paper, 2015. [Online]. Available: https://www.cisco.com/c/dam/enus/solutions/trends/iot/docs/computing-overview.pdf
[9] A. Tongaonkar, "A look at the mobile app identification landscape," *IEEE Internet Comput.*, vol. 20, no. 4, pp. 9–15, Jul. 2016.
[10] "Cisco global cloud index: Forecast and methodology," Cisco, San Jose, CA, USA, White Paper, 2016. [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/service-provider/global-cloud-index-gci/white-paper-c11-738085.html
[11] X. Chen, H. Liu, and D. Yang, "Improved LSH for privacy-aware and robust recommender system with sparse data in edge environment," *EURASIP J. Wireless Commun. Netw.*, vol. 2019, no. 1, p. 171, Dec. 2019, doi: 10.1186/s13638-019-1478-1.
[12] X. Su, G. Sperli, V. Moscato, A. Picariello, C. Esposito, and C. Choi, "An edge intelligence empowered recommender system enabling cultural heritage applications," *IEEE Trans Ind. Informat.*, vol. 15, no. 7, pp. 4266–4275, Jul. 2019.
[13] L. Qi, X. Zhang, W. Dou, C. Hu, C. Yang, and J. Chen, "A two-stage locality-sensitive hashing based approach for privacy-preserving mobile service recommendation in cross-platform edge environment," *Future Gener. Comput. Syst.*, vol. 88, pp. 636–643, Nov. 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167739X18301420
[14] S. Wang, Y. Zhao, L. Huang, J. Xu, and C.-H. Hsu, "QoS prediction for service recommendations in mobile edge computing," *J. Parallel Distrib. Comput.*, vol. 127, pp. 134–144, May 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S074373151730268X
[15] X. Wang, Y. Han, V. C. M. Leung, D. Niyato, X. Yan, and X. Chen, "Convergence of edge computing and deep learning: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, to be published.
[16] X. Wang, Y. Han, C. Wang, Q. Zhao, X. Chen, and M. Chen, "In-edge AI: Intelligentizing mobile edge computing, caching and communication by federated learning," *IEEE Netw.*, vol. 33, no. 5, pp. 156–165, Sep. 2019.
[17] B. Chen, J. Wan, A. Celesti, D. Li, H. Abbas, and Q. Zhang, "Edge computing in IoT-based manufacturing," *IEEE Commun. Mag.*, vol. 56, no. 9, pp. 103–109, Sep. 2018.
[18] E. Li, L. Zeng, Z. Zhou, and X. Chen, "Edge AI: On-demand accelerating deep neural network inference via edge computing," *IEEE Trans. Wireless Commun.*, vol. 19, no. 1, pp. 447–457, Jan. 2020.
[19] A. Felfernig, S. P. Erdeniz, M. Jeran, A. Akcay, P. Azzoni, M. Maiero, and C. Doukas, "Recommendation technologies for IoT edge devices," *Procedia Comput. Sci.*, vol. 110, pp. 504–509, Jan. 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1877050917313145
[20] L. Yao, Q. Z. Sheng, A. H. H. Ngu, and X. Li, "Things of interest recommendation by leveraging heterogeneous relations in the Internet of Things," *ACM Trans. Internet Technol.*, vol. 16, no. 2, pp. 9:1–9:25, Mar. 2016, doi: 10.1145/2837024.
[21] A. Felfernig, S. Polat-Erdeniz, C. Uran, S. Reiterer, M. Atas, T. N. T. Tran, P. Azzoni, C. Kiraly, and K. Dolui, "An overview of recommender systems in the Internet of Things," *J. Intell. Inf. Syst.*, vol. 52, no. 2, pp. 285–309, Apr. 2019, doi: 10.1007/s10844-018-0530-7.
[22] J. Wu, Z. Su, S. Wang, and J. Li, "Crowd sensing-enabling security service recommendation for social fog computing systems," *Sensors*, vol. 17, no. 8, p. 1744, Aug. 2017.
[23] Y. Cao, O. Kaiwartya, Y. Zhuang, N. Ahmad, Y. Sun, and J. Lloret, "A decentralized deadline-driven electric vehicle charging recommendation," *IEEE Syst. J.*, vol. 13, no. 3, pp. 3410–3421, Sep. 2019.
[24] I. Mashal, O. Alsaryrah, and T.-Y. Chung, "Testing and evaluating recommendation algorithms in Internet of Things," *J. Ambient Intell. Humanized Comput.*, vol. 7, no. 6, pp. 889–900, Dec. 2016, doi: 10.1007/s12652-016-0357-4.
[25] G. White, A. Palade, and S. Clarke, "QoS prediction for reliable service composition in IoT," in *Proc. Service-Oriented Comput. ICSOC Workshops*. Cham, Switzerland: Springer, Jun. 2018, pp. 149–160.
[26] Y. Yin, L. Chen, Y. Xu, J. Wan, H. Zhang, and Z. Mai, "Qos prediction for service recommendation with deep feature learning in edge computing environment," *Mobile Netw. Appl.*, Apr. 2019, doi: 10.1007/s11036-019-01241-7.
[27] Y. Zhang, M. S. Hossain, A. Ghoneim, and M. Guizani, "COCME: Content-oriented caching on the mobile edge for wireless communications," *IEEE Wireless Commun.*, vol. 26, no. 3, pp. 26–31, Jun. 2019.
[28] H. Zhu, Y. Cao, W. Wang, T. Jiang, and S. Jin, "Deep reinforcement learning for mobile edge caching: Review, new features, and open issues," *IEEE Netw.*, vol. 32, no. 6, pp. 50–57, Nov. 2018.
[29] Z. Chang, L. Lei, Z. Zhou, S. Mao, and T. Ristaniemi, "Learn to cache: Machine learning for network edge caching in the big data era," *IEEE Wireless Commun.*, vol. 25, no. 3, pp. 28–35, Jun. 2018.
[30] X. Wang, L. T. Yang, X. Xie, J. Jin, and M. J. Deen, "A cloud-edge computing framework for cyber-physical-social services," *IEEE Commun. Mag.*, vol. 55, no. 11, pp. 80–85, Nov. 2017.
[31] L. Yao, X. Wang, Q. Z. Sheng, S. Dustdar, S. Zhang, and S. Dustdar, "Recommendations on the Internet of Things: Requirements, challenges, and directions," *IEEE Internet Comput.*, vol. 23, no. 3, pp. 46–54, May 2019.
[32] T. Joachims, A. Swaminathan, Y. Raimond, O. Koch, and F. Vasile, "REVEAL 2018: Offline evaluation for recommender systems," in *Proc. 12th ACM Conf. Recommender Syst. (RecSys)*. New York, NY, USA: ACM, Sep. 2018, pp. 514–515, doi: 10.1145/3240323.3240334.
[33] Z.-S. Chen, J.-S.-R. Jang, and C.-H. Lee, "A kernel framework for content-based artist recommendation system in music," *IEEE Trans. Multimedia*, vol. 13, no. 6, pp. 1371–1380, Dec. 2011.
[34] M. Tkalcic, A. Odic, A. Kosir, and J. Tasic, "Affective labeling in a content-based recommender system for images," *IEEE Trans. Multimedia*, vol. 15, no. 2, pp. 391–400, Feb. 2013.
[35] S. Li, J. Wen, F. Luo, M. Gao, J. Zeng, and Z. Y. Dong, "A new QoS-aware Web service recommendation system based on contextual feature recognition at server-side," *IEEE Trans. Netw. Service Manage.*, vol. 14, no. 2, pp. 332–342, Jun. 2017.
[36] R. Chen, Q. Hua, Y.-S. Chang, B. Wang, L. Zhang, and X. Kong, "A survey of collaborative filtering-based recommender systems: From traditional methods to hybrid methods based on social networks," *IEEE Access*, vol. 6, pp. 64301–64320, 2018.
[37] R. L. Rosa, G. M. Schwartz, W. V. Ruggiero, and D. Z. Rodriguez, "A knowledge-based recommendation system that includes sentiment analysis and deep learning," *IEEE Trans Ind. Informat.*, vol. 15, no. 4, pp. 2124–2135, Apr. 2019.
[38] M. Gil, R. El Sherif, M. Pluye, B. C. M. Fung, R. Grad, and P. Pluye, "Towards a knowledge-based recommender system for linking electronic patient records with continuing medical education information at the point of care," *IEEE Access*, vol. 7, pp. 15955–15966, 2019.
[39] E. Amolochitis, I. T. Christou, and Z.-H. Tan, "Implementing a commercial-strength parallel hybrid movie recommendation engine," *IEEE Intell. Syst.*, vol. 29, no. 2, pp. 92–96, Mar. 2014.
[40] X.-L. Zheng, C.-C. Chen, J.-L. Hung, W. He, F.-X. Hong, and Z. Lin, "A hybrid trust-based recommender system for online communities of practice," *IEEE Trans. Learn. Technol.*, vol. 8, no. 4, pp. 345–356, Oct. 2015.
[41] P. R. Kumar, P. H. Raj, and P. Jelciana, "Exploring data security issues and solutions in cloud computing," *Procedia Comput. Sci.*, vol. 125, pp. 691–697, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1877050917328570
[42] Z. Tari, "Security and privacy in cloud computing," *IEEE Cloud Comput.*, vol. 1, no. 1, pp. 54–57, May 2014.
[43] F. Aznoli and N. J. Navimipour, "Cloud services recommendation: Reviewing the recent advances and suggesting the future research directions," *J. Netw. Comput. Appl.*, vol. 77, pp. 73–86, Jan. 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1084804516302375
[44] B. Cao, L. Zhang, Y. Li, D. Feng, and W. Cao, "Intelligent offloading in multi-access edge computing: A State-of-the-Art review and framework," *IEEE Commun. Mag.*, vol. 57, no. 3, pp. 56–62, Mar. 2019.
[45] Z. Zhou, X. Chen, E. Li, L. Zeng, K. Luo, and J. Zhang, "Edge intelligence: Paving the last mile of artificial intelligence with edge computing," *Proc. IEEE*, vol. 107, no. 8, pp. 1738–1762, Aug. 2019.
[46] M. Chan, E. Campo, D. Estève, and J.-Y. Fourniols, "Smart homes—Current features and future perspectives," *Maturitas*, vol. 64, no. 2, pp. 90–97, Oct. 2009. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0378512209002606

[47] R. Trimananda, A. Younis, B. Wang, B. Xu, B. Demsky, and G. Xu, "Vigilia: Securing smart home edge computing," in *Proc. IEEE/ACM Symp. Edge Comput. (SEC)*, Oct. 2018, pp. 74–89.

[48] S. Zhou and L. Zhang, "Smart home electricity demand forecasting system based on edge computing," in *Proc. IEEE 9th Int. Conf. Softw. Eng. Service Sci. (ICSESS)*, Nov. 2018, pp. 164–167.

[49] J. Cao, Q. Zhang, and W. Shi, "EdgeOSH: A home operating system for Internet of everything," in *Proc. IEEE Int. Conf. Distrib. Comput. Syst.*, Jun. 2017, pp. 1756–1764.

[50] C. Carr and M. Hesse, "Some notes on smart cities and the corporatization of urban governance," in *Proc. Int. Symp. Emergence Smart City*, Luxembourg City, Luxembourg, Mar. 2019.

[51] Y.-C. Hsieh, H.-J. Hong, P.-H. Tsai, Y.-R. Wang, Q. Zhu, M. Y. S. Uddin, N. Venkatasubramanian, and C.-H. Hsu, "Managed edge computing on Internet-of-Things devices for smart city applications," in *Proc. NOMS-IEEE/IFIP Netw. Oper. Manage. Symp.*, Apr. 2018, pp. 1–2.

[52] Y. de Bock, B. Braem, D. Subotic, M. Weyn, and J. M. Marquez-Barja, "Demo abstract: Crowd analysis with infrared sensor arrays on the smart city edge," in *Proc. IEEE Conf. Comput. Commun. Workshops (INFOCOM WKSHPS)*, Apr. 2019, pp. 983–984.

[53] Z. Li, X. Zhou, and Y. Qin, "A survey of mobile edge computing in the industrial Internet," in *Proc. 7th Int. Conf. Inf., Commun. Netw. (ICICN)*, Apr. 2019, pp. 94–98.

[54] P. Radanliev, D. C. De Roure, J. R. Nurse, R. M. Montalvo, and P. Burnap, "The industrial Internet-of-Things in the industry 4.0 supply chains of small and medium sized enterprises," Univ. Oxford, Oxford, U.K., Tech. Rep., 2019.

[55] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2322–2358, 4th Quart., 2017.

[56] D. Sabella, A. Vaillant, P. Kuure, U. Rauschenbach, and F. Giust, "Mobile-edge computing architecture: The role of MEC in the Internet of Things," *IEEE Consum. Electron. Mag.*, vol. 5, no. 4, pp. 84–91, Oct. 2016.

[57] P. Mach and Z. Becvar, "Mobile edge computing: A survey on architecture and computation offloading," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 3, pp. 1628–1656, 3rd Quart., 2017.

[58] H. Zhou, H. Wang, X. Li, and V. C. M. Leung, "A survey on mobile data offloading technologies," *IEEE Access*, vol. 6, pp. 5101–5111, 2018.

[59] J. Zheng, Y. Cai, Y. Wu, and X. Shen, "Dynamic computation offloading for mobile cloud computing: A stochastic game-theoretic approach," *IEEE Trans. Mobile Comput.*, vol. 18, no. 4, pp. 771–786, Apr. 2019.

[60] D. Li, Y. Han, C. Wang, G. Shi, X. Wang, X. Li, and V. C. M. Leung, "Deep reinforcement learning for cooperative edge caching in future mobile networks," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Apr. 2019, pp. 1–6.

[61] X. Li, X. Wang, P.-J. Wan, Z. Han, and V. C. M. Leung, "Hierarchical edge caching in device-to-device aided mobile networks: Modeling, optimization, and design," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 8, pp. 1768–1785, Aug. 2018.

[62] X. Li, X. Wang, K. Li, and V. C. M. Leung, "CaaS: Caching as a service for 5G networks," *IEEE Access*, vol. 5, pp. 5982–5993, 2017.

[63] C. Wang, S. Wang, D. Li, X. Wang, X. Li, and V. C. M. Leung, "Q-learning based edge caching optimization for D2D enabled hierarchical wireless networks," in *Proc. IEEE 15th Int. Conf. Mobile Ad Hoc Sensor Syst. (MASS)*, Oct. 2018, pp. 55–63.

[64] T. Taleb, K. Samdanis, B. Mada, H. Flinck, S. Dutta, and D. Sabella, "On multi-access edge computing: A survey of the emerging 5G network edge cloud architecture and orchestration," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 3, pp. 1657–1681, 3rd Quart., 2017.

[65] F. Xia, N. Y. Asabere, A. M. Ahmed, J. Li, and X. Kong, "Mobile multimedia recommendation in smart communities: A survey," *IEEE Access*, vol. 1, pp. 606–624, 2013.

[66] Y. Tang, H. Wang, K. Guo, T. Luo, and T. Chi, "A new replica placement mechanism for mobile media streaming in edge computing," *Concurrency Comput., Pract. Exper.*, to be published. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/cpe.5361

[67] J. Yang, H. Wang, Z. Lv, W. Wei, H. Song, M. Erol-Kantarci, B. Kantarci, and S. He, "Multimedia recommendation and transmission system based on cloud platform," *Future Gener. Comput. Syst.*, vol. 70, pp. 94–103, May 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167739X1630200X

[68] W. He, Y. Su, X. Xu, Z. Luo, L. Huang, and X. Du, "Cooperative content caching for mobile edge computing with network coding," *IEEE Access*, vol. 7, pp. 67695–67707, 2019.

[69] Y. Cao, N. Wang, G. Kamel, and Y.-J. Kim, "An electric vehicle charging management scheme based on Publish/Subscribe communication framework," *IEEE Syst. J.*, vol. 11, no. 3, pp. 1822–1835, Sep. 2017.

[70] F. Lin, Y. Zhou, I. You, J. Lin, X. An, and X. Lu, "Content recommendation algorithm for intelligent navigator in fog computing based IoT environment," *IEEE Access*, vol. 7, pp. 53677–53686, 2019.

[71] Y. Kun, W. Xiao-Ling, and Z. Ao-Ying, "Underlying techniques for Web services: A survey," *J. Softw.*, vol. 15, no. 3, pp. 428–442, Mar. 2004.

[72] Y. Li and Y. Guo, "Cultural distance-aware service recommendation approach in mobile edge computing," *Sci. Program.*, vol. 2018, pp. 1–8, Feb. 2018.

[73] Y. Zhou, Z. Tang, L. Qi, X. Zhang, W. Dou, and S. Wan, "Intelligent service recommendation for cold-start problems in edge computing," *IEEE Access*, vol. 7, pp. 46637–46645, 2019.

[74] F. Colace, M. D. Santo, L. Greco, V. Moscato, and A. Picariello, "A collaborative user-centered framework for recommending items in online social networks," *Comput. Hum. Behav.*, vol. 51, pp. 694–704, Oct. 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0747563214007262

[75] R. M. Frey, R. Xu, and A. Ilic, "A novel recommender system in IoT," in *Proc. Int. Conf. Internet Things*, Oct. 2015, pp. 26–28.

[76] W. Gong, L. Qi, and Y. Xu, "Privacy-aware multidimensional mobile service quality prediction and recommendation in distributed fog environment," *Wireless Commun. Mobile Comput.*, vol. 2018, pp. 1–8, Apr. 2018.

[77] W. Jiang, R. Wang, Z. Xu, Y. Huang, S. Chang, and Z. Qin, "PRUB: A privacy protection friend recommendation system based on user behavior," *Math. Problems Eng.*, vol. 2016, pp. 1–12, Aug. 2016.

[78] Z. Liu, Q. Z. Sheng, W. E. Zhang, D. Chu, and X. Xu, "Context-aware multi-QoS prediction for services in mobile edge computing," in *Proc. IEEE Int. Conf. Services Comput. (SCC)*, Jul. 2019, pp. 72–79.

[79] G. White, A. Palade, C. Cabrera, and S. Clarke, "Autoencoders for QoS prediction at the edge," in *Proc. IEEE Int. Conf. Pervas. Comput. Commun. (PerCom)*, Mar. 2019, pp. 1–9.

[80] P. Zhou, K. Wang, J. Xu, and D. Wu, "Differentially-private and trustworthy online social multimedia big data retrieval in edge computing," *IEEE Trans. Multimedia*, vol. 21, no. 3, pp. 539–554, Mar. 2019.

[81] L. Qi, X. Zhang, W. Dou, and Q. Ni, "A distributed locality-sensitive hashing-based approach for cloud service recommendation from multi-source data," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 11, pp. 2616–2624, Nov. 2017.

[82] L. Qi, R. Wang, C. Hu, S. Li, Q. He, and X. Xu, "Time-aware distributed service recommendation with privacy-preservation," *Inf. Sci.*, vol. 480, pp. 354–364, Apr. 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0020025518309186

[83] C. Wu, Y. Zhang, J. Jia, and W. Zhu, "Mobile contextual recommender system for online social media," *IEEE Trans. Mobile Comput.*, vol. 16, no. 12, pp. 3403–3416, Dec. 2017.

[84] A. F. Subahi, "Edge-based IoT medical record system: Requirements, recommendations and conceptual design," *IEEE Access*, vol. 7, pp. 94150–94159, 2019.

[85] E. Kim, "User space customized recommendation service platform system in mobile edge environment," in *Proc. 10th Int. Conf. Ubiquitous Future Netw. (ICUFN)*, Jul. 2018, pp. 622–624.

[86] L. Pu, X. Chen, J. Xu, and X. Fu, "Content retrieval at the edge: A social-aware and named data cooperative framework," *IEEE Trans. Emerg. Topics Comput.*, vol. 7, no. 1, pp. 135–148, Jan. 2019.

[87] B. Bai, L. Wang, Z. Han, W. Chen, and T. Svensson, "Caching based socially-aware D2D communications in wireless content delivery networks: A hypergraph framework," *IEEE Wireless Commun.*, vol. 23, no. 4, pp. 74–81, Aug. 2016.

[88] F. Malandrino, C.-F. Chiasserini, and S. Kirkpatrick, "The impact of vehicular traffic demand on 5G caching architectures: A data-driven study," *Veh. Commun.*, vol. 8, pp. 13–20, Apr. 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2214209616300742

[89] P. Sermpezis, T. Giannakas, T. Spyropoulos, and L. Vigneri, "Soft cache hits: Improving performance through recommendation and delivery of related content," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 6, pp. 1300–1313, Jun. 2018.

[90] A. Nadembega, A. Hafid, and T. Taleb, "A destination and mobility path prediction scheme for mobile networks," *IEEE Trans. Veh. Technol.*, vol. 64, no. 6, pp. 2577–2590, Jun. 2015.

[91] L. E. Chatzieleftheriou, M. Karaliopoulos, and I. Koutsopoulos, "Jointly optimizing content caching and recommendations in small cell networks," *IEEE Trans. Mobile Comput.*, vol. 18, no. 1, pp. 125–138, Jan. 2019.

[92] D. Liu and C. Yang, "A learning-based approach to joint content caching and recommendation at base stations," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2018, pp. 1–7.
[93] D. Liu and C. Yang, "A deep reinforcement learning approach to proactive content pushing and recommendation for mobile users," *IEEE Access*, vol. 7, pp. 83120–83136, 2019.
[94] K. Guo and C. Yang, "Temporal-spatial recommendation for caching at base stations via deep reinforcement learning," *IEEE Access*, vol. 7, pp. 58519–58532, 2019.
[95] K. Jung, B. Kulvatunyou, S. Choi, and M. P. Brundage, "An overview of a smart manufacturing system readiness assessment," in *Advances in Production Management Systems. Initiatives for a Sustainable World*. Cham, Switzerland: Springer, Sep. 2016, pp. 705–712.
[96] V. Subramaniyaswamy, G. Manogaran, R. Logesh, V. Vijayakumar, N. Chilamkurti, D. Malathi, and N. Senthilselvan, "An ontology-driven personalized food recommendation in IoT-based healthcare system," *J. Supercomput.*, vol. 75, no. 6, pp. 3184–3216, Jun. 2019.
[97] Z. Zhou, X. Chen, E. Li, L. Zeng, K. Luo, and J. Zhang, "Edge intelligence: Paving the last mile of artificial intelligence with edge computing," 2019, *arXiv:1905.10083*. [Online]. Available: http://arxiv.org/abs/1905.10083
[98] F. Li, K.-Y. Lam, X. Li, Z. Sheng, J. Hua, and L. Wang, "Advances and emerging challenges in cognitive Internet of Things," *IEEE Trans Ind. Informat.*, to be published.

CHUAN SUN (Student Member, IEEE) received the B.E. degree from the School of Information Science and Engineering, Wuhan University of Science and Technology, in July 2017. He is currently pursuing the Ph.D. degree with the School of Big Data and Software Engineering, Chongqing University. His current research interests include mobile edge computing, service computing, and machine learning.

HUI LI received the bachelor's degree in computer science and technology from Northeast Agricultural University, in July 2019. He is currently pursuing the master's degree with the School of Big Data and Software Engineering, Chongqing University. His current research interests include mobile edge computing, cloud computing, and big data.

XIUHUA LI (Member, IEEE) received the B.S. degree from the Honors School, Harbin Institute of Technology, Harbin, China, in 2011, the M.S. degree from the School of Electronics and Information Engineering, Harbin Institute of Technology, in 2013, and the Ph.D. degree from the Department of Electrical and Computer Engineering, The University of British Columbia, Vancouver, BC, Canada, in 2018. He joined Chongqing University, China, through the One-Hundred Talents Plan of Chongqing University, in 2019. He is currently a tenure-track Assistant Professor with the School of Big Data and Software Engineering, Chongqing University, and the Dean of the Institute of Intelligent Software and Services Computing associated with the Key Laboratory of Dependable Service Computing in Cyber Physical Society, Ministry of Education, Chongqing University. His current research interests include the 5G/B5G mobile Internet, mobile edge computing and caching, big data analytics, and machine learning.

JUNHAO WEN (Member, IEEE) received the Ph.D. degree from Chongqing University, in 2008. He is currently a Vice Head and a Professor with the School of Big Data and Software Engineering, Chongqing University. He has over 30 research and industrial grants and developed many commercial systems and software tools. He has published over 80 refereed journal articles and conference papers in these areas. His research interests include service computing, edge computing, and software dependable engineering.

QINGYU XIONG (Member, IEEE) received the B.Sc. and M.Sc. degrees from Chongqing University, in 1986 and 1991, respectively, and the Ph.D. degree from Kyushu University, Japan, in 2002. He is currently a Professor and a Supervisor for the Ph.D. student with Chongqing University. His main research interests include intelligent system and intelligent computation and pervasive computation.

WEI ZHOU received the Ph.D. degree from the School of Computer Science, Chongqing University, in 2014. He is currently a Professor with the School of Big Data and Software Engineering, Chongqing University. His research interests include data mining, machine learning, and recommender systems.