Future Generation Computer Systems 102 (2020) 66–74

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

Cognitive multi-agent empowering mobile edge computing for resource caching and collaboration

Rui Wang [a], Miao Li [a], Limei Peng [b,*], Ying Hu [c,*], Mohammad Mehedi Hassan [d,e], Abdulhameed Alelaiwi [d,e]

[a] School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan 430074, China
[b] School of Computer Science and Engineering, Kyungpook National University, Daegu 41566, South Korea
[c] School of Mathematics and Economics, Hubei University of Education, Wuhan 430205, China
[d] Research Chair of Smart Technologies, King Saud University, Riyadh 11543, Saudi Arabia
[e] College of Computer and Information Sciences, King Saud University, Riyadh 11543, Saudi Arabia

HIGHLIGHTS

- Cognitive agent (CA) is put forward in this article to realize cognition function on mobile edge computing (MEC) in smart devices aiming at fifth generation of the mobile communication system. In detail, CA is used to build personalized model of users' behavior data.
- The prediction model of users' behavior trajectory and application based on the aware ability of CA, show the cognition of agent. Based on the predicting result, the caching strategy and cache business are generated to shorten the delay of task execution.
- To verify the effectiveness of CA, a model is formulated about resource caching and collaboration including hit rate and delay of task execution.
- Simulation experiments are performed to prove that CA can well recognize and help terminal equipment execute resource caching and collaboration and the quality of experience and service are improved.

ARTICLE INFO

Article history:
Received 16 July 2019
Received in revised form 31 July 2019
Accepted 1 August 2019
Available online 6 August 2019

Keywords:
Cognitive agent
Mobile edge computing
Caching strategy
Resource collaboration

ABSTRACT

The service of mobile network develops rapidly nowadays, which generates various computing and resource-intensive applications, such as Internet of vehicles and virtual reality. Mobile edge computing (MEC) is close to data source and users, so terminals can execute tasks at the edge of network. In this way, the heavy load on core network can be relieved and tasks can be executed effectively. However, the demands of users vary from each other and users move all the time. It is difficult for the existing way of service supply to meet demands of all users. Cognitive Agent (CA) is put forward in this paper to help users cache and execute tasks on MEC in advance. In detail, CA is used to build personalized model combined with users' behavior data. At the same time, it uses Long short-term memory neural network to forecast the moving trajectory of terminal equipment and the service types to be requested, uses the prediction result to generate caching strategy, cache business and shorten the delay of task execution. Besides, to further reduce the stress on MEC, we propose the collaboration of computing, communicating and caching resource with neighboring users' equipment. To verify the effectiveness of CA, we build a model that assesses the performance of the system. Finally, we design a simulation experiment to execute resource request and resource collaboration. The result of the experiments show that CA can improve the efficiency of communication network, relieve the stress on network and improve the quality of services to users.

1. Introduction

At present, emerging information services and applications increase rapidly with the development of wireless communication and Internet of Things (IoT) [1,2]. More and more intelligent devices, such as wearable devices, remote medical system and virtual/augmented reality (VR/AR), need advanced communication technology and computing ability to guarantee ultra-reliable low-latency communication (URLLC), which is a great challenges

* Corresponding author.
E-mail addresses: ruiwang2018@hust.edu.cn (R. Wang), miaoli@hust.edu.cn (M. Li), auroraplm@knu.ac.kr (L. Peng), yinghu@hust.edu.cn (Y. Hu), mmhassan@ksu.edu.sa (M.M. Hassan), aalelaiwi@ksu.edu.sa (A. Alelaiwi).

https://doi.org/10.1016/j.future.2019.08.001
0167-739X/

for network suppliers [3,4]. Besides, wireless access device in large quantity outputs massive quantity of data all the time [5]. The reasonable use of valuable information is greatly significant for improving the quality of services supplied to users [6].

Mobile edge computing (MEC) supplies infrastructures of communication, computing and caching at the network edge that is close to data source or users [7]. It offers cloud services and information technology environment for edge applications and supplies high-quality communication services to users [8–13]. The advantages of the emerging service model MEC can be summarized below: (i). Due to be close to users or data source, some problems can be solved, such as long delay and heavy data traffic. It well supports the real-time and bandwidth-dense Internet of Things facilities, especially for the emotion-aware multimedia systems [14–16]. Taking unmanned driving for example, automobile needs to calculate running speed and the distance from others and sense surroundings in real time. Data transmission with short delay can effectively decrease the probability of traffic accidents. (ii). MEC deploys resources based on edge network caching technology, which decreases the enormous waste of resource while transmitting the same content repeatedly on the internet [17,18]. At the same time, the delay of transmission can be largely shortened. Based on content popularity, ultra high definition video can be cached and asynchronous content multiplexing can be realized [19].

A collaborative caching architecture based on edge caching is put forward in [20]. MEC resource is used to improve edge caching ability. In result, it can minimize delay of content accessing and increase the rate of using caching resources. Luo et al. consider the energy consumption, backhaul capacity and content popularity distribution of MEC server and a joint optimization framework is built to minimize the energy consumption of the system and find the best caching position to achieve higher energy efficiency [21]. In the above schemes, it is considered to request data and project caching strategy for multi-users to achieve better performance but demand of single user is ignored. Personalized services cannot be supplied to them [22,23]. At network edge, the resources requested by users are greatly different because of the difference of context [24]. Context includes network status, such as personal data of users and spatial–temporal data. Zeng et al. put forward a heuristic intelligent caching algorithm [25], where users' behaviors are simulated to reflect their different demands and on the basis of low cost, high caching hit rate and stability can be realized. However, there is little resource on edge cloud in general. The resources at edge are especially scarce, which indicates it is impossible to store the personal behavior model of all users on edge cloud. In other words, the reference for caching content cannot be offered to edge cloud in real time. Considering the problem,Ren et al. research the cooperation between cloud computing and edge computing [26], which cannot still solve the problem of scarce resources on edge cloud [27,28]. As for some IoT devices, especially Internet of Vehicles, they have high mobility, which means there are several edge clouds within short time and even the switch among clouds [29,30]. It greatly adds to the difficulty of getting users' data and behavior models [31,32].

Agent is a computer program module that can sense context, play functions sustainably and independently and cope with the change of environment [33]. With interaction and initiative, the module can work independently, which has been applied in many fields [34]. Daniel et al. use the mobile algorithm based on Agent [35]. Miniature unmanned aerial vehicle can carry sensing device in severe high environment to achieve high-quality communication at high coverage rate. Aiming at VANETs, Premkumar et al. build cognitive agent and put forward a stable routing protocol to seek the most stable route of information transmission [36]. However, the Agent mentioned above has no cognition ability. It cannot apply personalized data to generate optimal scheme for users [37,38].

Based on the above discussions, this paper introduces cognitive agent (CA) realizing cognition function on MEC in smart devices aiming at the fifth generation of the mobile communication system (5G). Every smart device owning a CA can perceive context, store the application data and communication data of IoT devices and generate the behavior pattern of users' device. At the same time, it can serve users' terminal, respond to the change of network environment and business demand and enhance users' quality of experience (QoE) and quality of service (QoS) [39]. To sum up, the main contributions of this paper are listed below:

(1) It puts forward the idea that CA makes cognition ability to users' device and relieves the burden on MEC. Users' personalized model is generated on the basis of users' data. The storage and computing pressure on MEC can be decreased. Besides, MEC can be further intelligentized.

(2) It puts forward the prediction model of users' behavior trajectory and application based on the sensory ability of CA, showing the cognition of agent. In addition, MEC can cache corresponding resources according to the prediction result, which greatly increase the efficiency of caching.

(3) It improves the ability of resource collaboration among users' device based on the interaction of CA, which decrease the waste of resource. By applying the interaction among different users' device agents, the collaboration of communication, computing and caching resources can be realized in case of enough network resources.

The remainder of this paper is organized as follows: Section 2 introduces the researches on agent and MEC. Section 3 presents the framework of CA on edge cloud. Taking Internet of Vehicles for example, design details are introduced. Section 4 builds prediction model to realize high efficiency of caching and low energy consumption of communication. Section 5 carries out relevant experiments to prove the performance of CA. Finally, Section 6 summarizes the whole paper.

2. Related work

The prominent research questions of the article is to use CA to predict users' required services in position and time series. This is for caching business on edge cloud and the collaboration of communication, computing and caching resources among different users' device. Related issues have also been elaborated in recent studies [40].

Caching technology is a means that uses space to get more time. The locality contributes to its effectiveness, including space locality and time locality. Generally, research in two directions can be involved in the research on caching, including what to cache [41–45] and when to cache [46–48]. The edge content caching is based on popularity. In detail, it can be divided to be the caching based on model [42] and the caching based on learning [43–45]. Abdallah Khreishah et al. [42] designed complete multinomial time approximation algorithm to research the caching of collaboration content among base stations, which minimized the total operating costs. Zhi Wang et al. [43] researched the relevance between the video propagation mode in microblog system and the video popularity on video sharing website. A learning framework based on neural network is designed to predict the number of potential audiences and their geographical distribution. S. Li et al. [44]introduced a caching replacement method PopCaching. Cache hit ratio was greatly raised by means of online popularity learning. Edge caching time strategy can

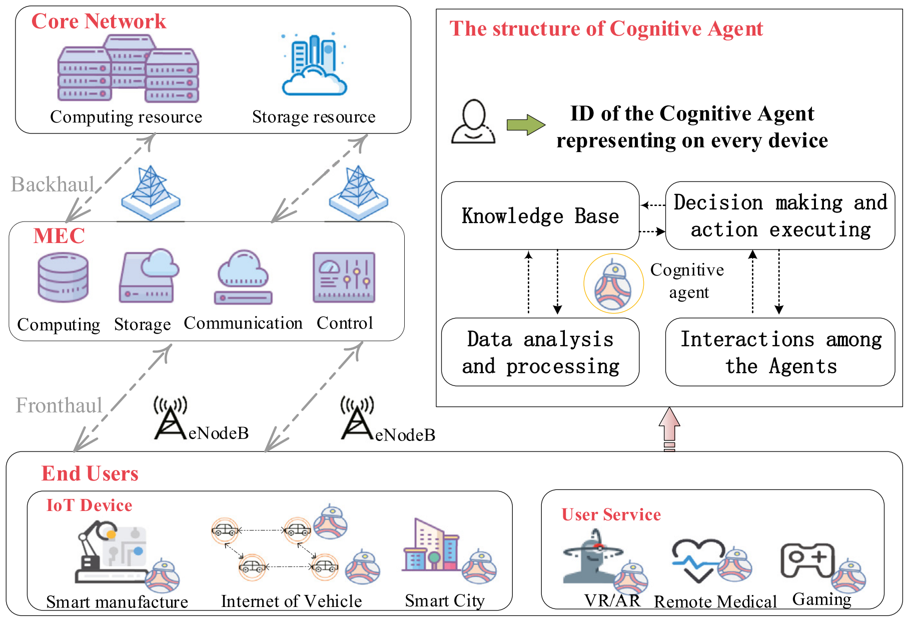


Fig. 1. The framework of the cognitive agent paradigm.

be proactive caching [46] and adaptive caching [47,48]. Shen et al. [46] used the stackelberg model in game theory and operators stimulated caching based on the price of traffic in peak period of business. Mobility of users was considered in [47] to put forward adaptive caching strategy. Tanzil et al. [48] proposeed an adaptive caching strategy with content sensing and cellular network sensing. Based on the above and the historical data in CA, a users' application behavior model is built on edge cloud by active learning. Then proactive caching and resource collaboration is combined on the basis of users' mobility trajectory.

Agent is an entity of cognition distributed in environment for sensing environment, exchanging information and executing decisions. The interaction of CA can be used to realize the collaboration and combination of communication, computing and caching resources on different users' devices in certain area. Xiaofei Wang et al. [39] designed edge AI framework to combine deep reinforcement learning framework and associative learning framework with mobile edge system and optimize the communication, computing and caching on mobile edge. M. Chen et al. [49] put forward the new concept of computing task caching Edge-CoCaCo namely the joint optimization of computing, caching and communication on edge cloud. Yaping Sun et al. [50] introduced a mobile VR delivery framework based on MEC and revealed the balance among communication, computing and caching. Anselme Ndikumana et al. [51] applied a block successive upper bound minimization method to solve the problems among communication, computing, caching and control. This was to reach the goals of minimizing delay and maximizing saving of bandwidth. The research of them show the resource collaboration on MEC. In this paper, CA can be applied to unite different device for interaction. This is to decrease the consumption of communication, caching and computing resources in communication network and realize the network of URLLC.

Therefore, user personalization model built by CA is used in this paper to predict users' application use behaviors and then cache relevant content on MEC. At the same time, the collaboration of communication, computing and caching resources minimizes consumption of energy and stimulates the interaction among different IoT device.

3. Proposed the cognitive agent paradigm and design issues

We introduce the agent paradigm in mobile edge computing in this part. Taking the resource caching and collaboration in Internet of Vehicles for example, it is expounded in detail that CA helps MEC carry out content caching and the collaboration of communication, computing and caching resources in Internet of Vehicles.

3.1. The design of the cognitive agent paradigm

CA is a computer program module with cognition ability that depends on terminal users' device and helps users request resources and interact resources. It is mainly responsible for users' resource management on MEC. The paradigm framework of CA is shown in Fig. 1.

This paper mainly focuses on the three layers of network. In details, terminal users request the resources on MEC by wireless network and MEC requests resources from upper core network by base station. The unified database in core network contains dynamic strategy data, semi-static user data and static network data. Network controls strategy correspondingly based on the data. MEC has some computing resources and caching resources. At the same time, it controls resources (such as resource caching and unloading) and manages communication resources. As for terminal users, different device accesses wireless network, such as the devices of IoT, intelligent manufacturing, Internet of Vehicles and smart city which highly require delay and the services which highly require computing performance, such as VR/AR, remote medical system, gaming etc. To meet the expectation of intelligent devices, CA supplies personalized model for users to meet their requirement on URLLC. The four functional modules of CA are introduced below:

- Knowledge Base (KB): It includes users' personal historical data and personalized model. Because the CA is for each user device, different data is generated for different devices, including application data and historical data in time and space sequences, so each agent is responsible for recording the data generated by its device and storing the user personalized model obtained from the data. Besides, KB contains the application environment and network environment sensed by agent. KB is the basis of executing the other three functional modules. Based on the information stored, the status and use condition of the device can be known well.

- Decision making and action executing: CA helps users request resources, execute caching strategy and synergize with the resources of other agents. Caching strategy can be made by predicting users' behavior track and the use of application contents in KB. By requesting the resources on MEC at time and position nodes in advance, resources can be cached and transmitted easily. At the same time, if the resources requested by one device can be supplied by the neighboring device, this module is responsible for resource collaboration with other agents to complete the transmission of resources.
- Data analysis and processing: Online learning and incremental learning are applied to the data collected to update the information in KB. Incremental Learning refers to a learning system that continually learns new knowledge from new samples and preserves most of the knowledge that has been learned before. CA can perceive the application environment of device, communication environment and the resource environment on MEC. These environmental information will be offered to the module for data analysis and processing. Users' future request of resources in time and space series can be predicted. Then it is transmitted to KB module to guarantee decisions can be made and actions can be taken.
- Interactions among the agents: CA exchanges information with the neighboring CAs for the resources collaboration. When different users who are close to each other request the same resources from MEC, network resources in large quantity will be consumed. Therefore, devices may request resources from the surrounding device directly to complete transmission of resources, which decreases the transmission delay on network and the computing of MEC. This module is responsible for sharing and exchanging information with neighboring agents and resources collaboration can be completed in permitted range of distance.

The main modules of CA are introduced above. Taking Internet of Vehicles for example, the functions of agent are described aiming at resource caching and collaboration.

3.2. Single-agent assisted caching on MEC

Internet of Vehicles is a specific scenario of IoT, which corresponds to diversified services. Terminal users and applications based on Vehicle to X(V2X) in large quantity exist in this scenario. The communication between vehicle and vehicle, vehicle and base station, base station and base station, vehicle and other objects can be realized by optimal channel access technologies [52]. As for Internet of Vehicles-oriented application scenario, LTE network framework based on MEC can shorten end-to-end delay of Uu mode by decreasing routing nodes of data transmission. Besides, the character of MEC coverage can be used to deploy services of Internet of Vehicles with geographical and regional features. To further shorten users' request delay and supply personalized services of Internet of Vehicles, content can be cached on edge cloud in advance. When user requests content, the content requested can be got by the user directly if the content is on edge cloud. This shortens the delay of requesting content, relieves the pressure of core network and improves users' QoE. The caching execution assisted by single-agent on edge cloud is in three stages introduced below.

- Online learning predicts caching content in time and space series. Based on the historical trajectory data of vehicles in KB of CA, the mobile model of users and service request model in given timestamp and position can be got. When vehicle moves, above model can be used to predict destination and route of the vehicle and the type of service to be requested utilized for caching in advance.
- The initiation of a cache request. After predicting the businesses to be requested by vehicle, CA is responsible for requesting business from the MEC of corresponding route. This is to cache service in advance and shorten delay of service. Besides, it is necessary to consider the switch of vehicle among RSU at different running speed to guaranteed to get resources from the nearest edge cloud.
- Incremental learning updates KB. The prediction of vehicles' behavior and the actual status of vehicles will return to KB. If the result of prediction is consistent with the actual condition, positive sample can be offered for updating KB model. On the contrary, it is negative sample. These contribute to building better personalized model and greatly improve users' experience.

3.3. Multi-agent enabled resource collaboration on MEC

There are different ways of classification for Internet of Vehicles in different perspectives. For example, its applications can be divided to be information service application that focuses on users' experiencing, the automobile intelligence application focusing on driving and intelligent transportation application focusing on collaboration. Information service application mainly involves interactive experience of vehicle owner, including guidance, entertainment and remote medical treatment. Automobile intelligence application involves intelligent traffic lights, adaptive cruise enhancement and intelligent parking management. Intelligent transportation application is based on wireless communication and sensor detection technology supporting city brain to treat the collaboration of city operation and governance intelligently [53]. Above applications need the great collaboration among vehicles, roads and environment. While meeting users' application need, system resources can be optimized at the same time.

Multi-agents enabled resource collaboration aims at the scarce capacity of user terminals and MEC which cannot meet demand of resources. Therefore, the collaboration between MEC servers and the computing, communication and resource caching of user terminal is considered to meet business demand. In other words, the terminal meets service request by computing, communication and caching. The purpose of resource collaboration is to improve users' quality of experience (QoE) and quality of service (QoS) of network supplier. Delay of terminal and total consumption of resource including computing energy consumption and transmission energy consumption will be used to signify them.

In Internet of Vehicles, one vehicle shares and collaborates resources when neighboring vehicle requests the same resources. The resources can be communication resource, computing resource and caching resource. In the process of driving, users in vehicle indulge themselves in experiencing game. Meanwhile, the neighboring users are playing the game in the same scenario, the resources of which have been cached on edge cloud. If the collaboration of resources is not considered, MEC will transmit the same data packet to the applications in Internet of Vehicles respectively, which means to occupy certain bandwidth of communication. In case of resource collaboration, CA will perceive the same game application of two devices when sharing and exchanging information and terminals could communicate

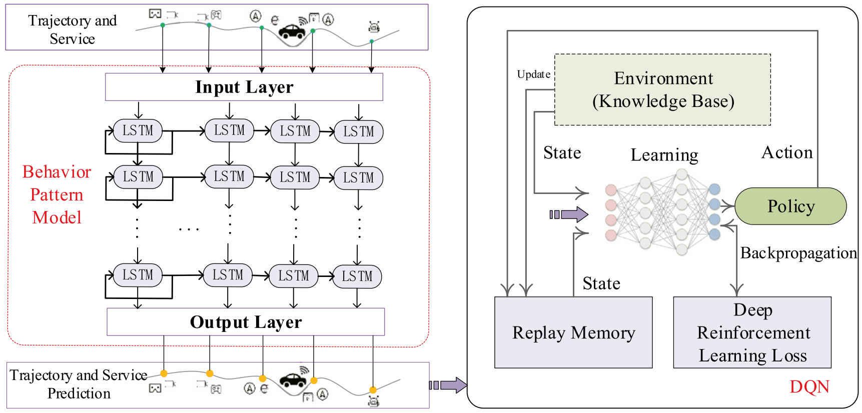


Fig. 2. The operation mechanism of cognitive agent.

to each other. The users who have the resource can transmit it to other users by D2D or other advanced cooperative communication technologies like [54]. The computing resource at MEC terminal is transformed to be the communication resource of terminals. Besides, the similar resource collaboration can be the transformation from communication resource to be caching resource and the transformation from computing resource to be caching resource. The cooperation among different terminals will increase capacity of the whole system, shorten the delay of transmission, decrease the consumption of energy resource and greatly guarantee QoE and QoS.

4. Problem formulation and solution

CA is responsible for predicting users' behaviors and making caching strategy. At the same time, it interacts with other Agents to complete the collaboration of resources. The target of caching and resource collaboration will be expounded in this part. In addition, the key technologies used for making caching strategy will be described in detail.

4.1. Key problem and technology on caching

The first problem of caching on MEC is how to choose the position of caching. After the agent predicts the trajectory of users, the contents users will be employed can be cached on the nearest cloud. As shown in Fig. 2, we firstly discuss how to predict the track of users. There are many algorithms of prediction, such as decision-making tree in traditional algorithm of machine learning, support vector machine (SVM) and K-nearest neighbor (KNN) and linear regression. Data can be trained rapidly and realized easily in the algorithms. However, these algorithms are not suitable for predicting users' moving tracks because of their weak performance of capturing context information in time series. Long short term memory (LSTM) algorithm is used in this paper to predict users' moving tracks. Users' historical tracks are signified to be $\mathcal{H} = \{l_0, l_1, \dots, l_{j-1}\}$. l_i is the position of user under timestamp i. $S_i = \{r_1, r_2, \dots, r_u\}$ is the type of service used in position l_i. Our goal is to predict the K series of the next position $\{l_j, \dots, l_{j+k-1}\}$ and the service type $\{s_j, \dots, s_{j+k-1}\}$ requested. The simulated training process is divided into three stages. Firstly, fully connected layer is responsible for treating the service type in position series and corresponding position series. Each value is mapped to be multidimensional eigenvector. Then the series treated will be transmitted to the main treatment part of LSTM model. It is a deep stacked LSTM neural network. The output series of the former layer will be the input of the next layer. Finally, the fully connected layer is responsible for outputting predicted position $\{\widetilde{l_j}, \dots, \widetilde{l_{j+k-1}}\}$ and the service type $\{\widetilde{s_j}, \dots, \widetilde{s_{j+k-1}}\}$ in the position. Finally, mobility pattern of users and type of business request are saved in KB. Then the future tracks and businesses can be predicted on the basis of training parameters of network. With the generation of new data, incremental learning can be used to update model.

After predicting users' moving tracks and business request type, CA needs to request corresponding services from MEC in corresponding position. The goal of prediction is to raise the hit rate of caching. The success rate of caching is defined below:

$$x_{i,j}^{s} = \begin{cases} 1 & \text{Hit} \\ 0 & \text{Not hit} \end{cases} \tag{1}$$

where $x_{i,j}^{s}$ is the caching hit value of user j to business s under the timestamp i. It will equal 1 if successful. Otherwise, it will equal 0.

Therefore, we can get the caching hit rate of user j under the timestamp i:

$$HitRate = \frac{\sum_{j=1}^{p} x_{i,j}}{sum\_services} \times 100\% \tag{2}$$

where $sum\_services$ is the total business request of user j under the timestamp i.

After predicting the moving track of users' devices and the services to be requested, we will use the caching strategy assisted by Q-learning. Q-learning is value-based algorithm in reinforcement learning algorithm. Q(s, a) is the expectation from taking action in the state of certain moment. Then CA will get the corresponding return according to feedback of action. In the process of updating and iteration, the actions for getting the maximum earnings can be got according to Q value. In other words, it is to maximize hit rate and minimize execution delay. In Fig. 2, deep Q-Learning network is used and it contains five parts. Environment means the KB in CA, where all data can be got. Deep neural network is responsible for generating caching strategy. There are three sources of data for training neural network including Knowledge Base, Replay Memory and the prediction of LSTM. Loss function adjusts parameters of network by back-propagating. The predicted values of neural network are put in Policy to generate corresponding action and updated in Replay Memory. Replay Memory is responsible for managing actions and state of system. Bellman equation is used to update Q value:

$$Q(s, a) \leftarrow (1 - \alpha)Q(s, a) + \alpha[r + \gamma \max_{a'} Q'(s', a')] \tag{3}$$

where α is learning rate; γ is discount factor; r is reward function; s and a signify state and action respectively. State includes terminal devices (mobility, battery power, etc.) and the availability of resources on the edge cloud.

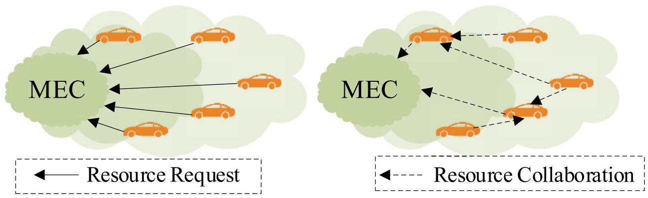


Fig. 3. The resource collaboration among vehicles.

Actions include whether to cache resources, whether to perform resource coordination, and so on. The ultimate goal of executing caching tasks is to shorten delay of task execution. Communication model of network shall be defined. According to Shannon's theorem, upstream data speed of system can be signified to be:

$$R_j = W \log_2 (1 + \frac{P_j \cdot H_j}{\omega}) \tag{4}$$

where W is the communication bandwidth of uplink; P_j is the transmission power of terminal; H_j is the gain of channel from user to base station; ω is the power of noise. As for local task computing model, f_j is the computing ability of each mobile terminal CPU; D_j is the number of cycles of CPU.

Therefore, local task computing time can be got:

$$T_j = \frac{D_j}{f_j} \tag{5}$$

Finally, the total time of task execution can be got:

$$\mathcal{T}_{i,j} = \begin{cases} T_j + \varepsilon_{i,j} & \text{Hit} \\ R_j + T_j + \varepsilon_{i,j} & \text{Not hit} \end{cases} \tag{6}$$

where $\varepsilon_{i,j}$ is the correction used for the interference factor incurred in the process of task execution.

4.2. System model of resource collaboration

Considering the application in the field of assisted driving, in order to ensure the safe communication between vehicles and ensure the minimum reachable rate, the CA performs information sharing and resource coordination, as shown in Fig. 3. D2D communication can be used to shorten the delay of transmission, increase speed of transmission and increase the devices carried by cell. A problem of resource collaboration is that several D2D users use multiple resources. Based on the premise of meeting users' demand, performance gain of users can be enhanced and performance loss of cellular users can be decreased. Performance loss of users can be reflected by the cost of reuse.

As for the resource collaboration in D2D network, the goal is to maximize handling capacity of network and minimize cost of reuse. Handling capacity of user's device is:

$$\mathcal{R}_d = W \log_2 (1 + \frac{P_d \cdot G_{d,d}}{\sigma^2 + P_u \cdot G_{u,d}}) \tag{7}$$

For the convenience of analysis, it is assumed that speed or emission power before reuse is same as that after reuse. In this way, performance loss is transformed to be speed loss of user. Performance loss can be described to be:

$$\mathcal{L}_d = W \log_2 (1 + \frac{\widetilde{P_u} \cdot G_{u,b}}{\sigma^2}) - W \log_2 (1 + \frac{P_u \cdot G_{u,b}}{\sigma^2 + P_d \cdot G_{d,b}}) \tag{8}$$

where P_d is user's emission power in D2D; P_u is the emission power of cellular user; $\widetilde{P_u}$ is the generation power of cellular user before reuse; σ^2 is the power of Gaussian noise. $G_{d,d}$ is the gain of link from D2D emission end to base station. $G_{u,d}$ is the gain of link from cell emission end to device receive end. $G_{d,b}$ is the gain of link from cell user to base station. And $G_{u,b}$ is the gain of link from cell to base station. Performance loss is the cost of multiplexing resource of cellular users. Assuming that after devices multiplex resource, the transmit power cannot change compared the before. And then the performance loss is the rate loss. Signal interference from the user device to the base station occurs after multiplexing the resources.

The goal of resource collaboration in this paper is to guarantee users' experiencing in D2D communication and relieve the communication and computing pressure at MEC.

Table 1
Parameter setting.

Parameters	Values
Time duration	100
The num of available server	15
The num of random and MRU	10
The way of Random	Randint (0, 10)
W, P_j, H_j, ω	2, 3, 4, 5
D_j, f_j	6,20
ε_i, j	Normal (1,0.5)
Cell link path loss	$128 + 37 \log_{10}$ (d/km)
D2D link path loss	$148 + 40 \log_{10}$ (d/km)
σ^2	60

5. Performance analysis

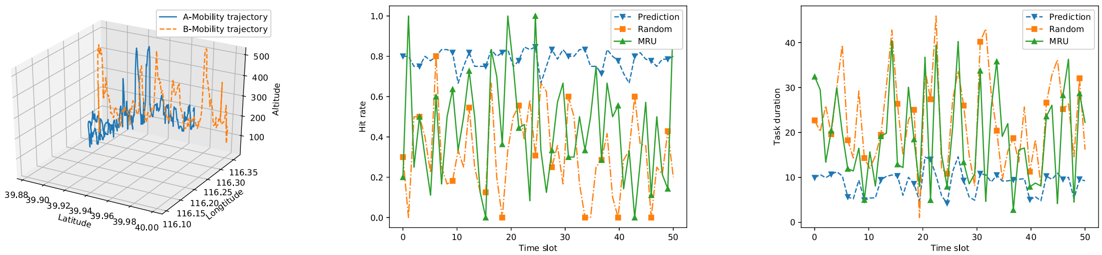
To reflect the cognition function of CA, its effectiveness after introducing it into MEC will be verified and relevant simulation experiments will be operated. In this paper, experiments in two aspects will be carried out. In one aspect, we predict the behavior pattern of users' devices aiming at their mobility and then execute relevant caching scheme to raise the hit rate of caching and shorten the delay of task execution. In the other aspect, D2D communication among device is verified. The collaboration among computing resource, communication resource and caching resource can increase the handling capacity of network, shorten the delay of task execution and effectively relieve the burden of MEC and the server of core network. The following assumption is used in this paper for targeted experiment. Base station can supply enough communication resources to MEC, when CA requests caching resource the delay of request will not be considered and base station can supply enough communication resources to users without waiting for allocation of resources. The detailed experiment process is introduced below:

5.1. Experiment setup

Data in different types are used in this paper for the experiment. To predict users' behaviors, we use real dataset to get the position of different users at different time. The dataset features include longitude, latitude, height and timestamp. As for the type of users' service request and the quantity of services requested, relevant experimental data can be got by simulation. It is note that link path loss is the link gain referred above and D2D link path loss is $G_{d,d}$. For details, please refer to Table 1.

5.2. The result of resource caching

As for the behavior track of users, the longitude, latitude and height of positions are used in this paper for measurement. Fig. 4(a) shows the users' mobile trajectory we used in the experiment, which is the point of a vehicle at different times. In the experiment, we assume that CA has got the vehicle trajectory point.



(a) The mobility trajectory of different times (b) The hit rate of different algorithms (c) The task duration of different algorithms

Fig. 4. The results of resource caching.

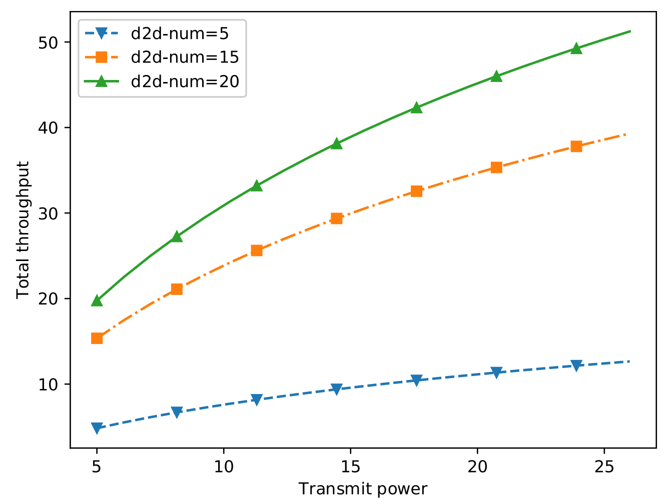
To choose caching strategy, we compare three algorithms: the caching based on prediction, random caching and the lately most frequently-used (MRU) caching. The caching based on prediction are realized by predict the business type from the give set. The MRU caching is the quantity of resources requested by users in the process of statistics task execution. In each round of choosing, the lately most frequently-used caching will be chosen. As for the quantity of predicted resources, we assume predicted caching is all the resource predicted. Random caching and MRU cache the 10 items of resource which meet requirements. In Fig. 4(b), the caching hit rate based on prediction fluctuates around 80%. Besides, it tends to be stable within 50 timestamps. The random caching strategy fluctuates greatly around 60%. The fluctuation amplitude of MRU algorithm is also large but its hit rate can reach 100% sometimes. In analysis, the reason is that MEC offers sufficient resources and the caching quantity for each user is not limited. In this experiment, it is set to 10. In actual system, this ideal cannot be achieved due to the limitation of the storage resource. As for the delay of task execution, the duration of caching algorithm tasks is short while the delay of other two algorithms is long. In above analyses by synthesis, the algorithm of CA for prediction can raise the hit rate of caching and effectively shorten delay, which shows that it is necessary to use agent.

5.3. The result of resource collaboration

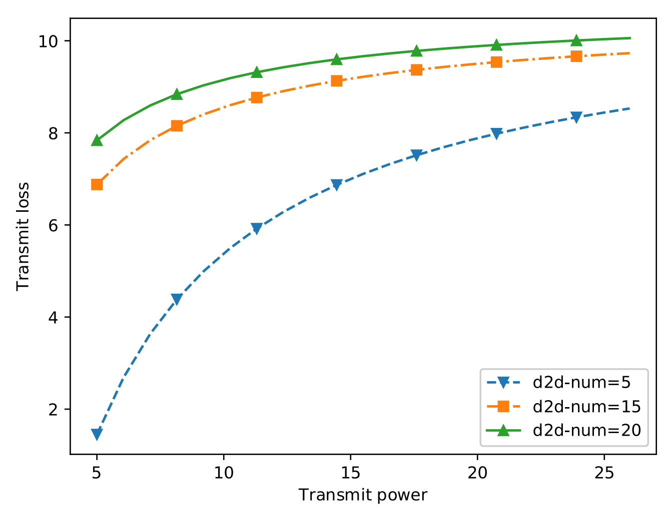
For resource collaboration, experiments are conducted in which multiple D2D users multiplex multiple cellular user resources. Considering the hybrid network where both cellular users and D2D users exist, D2D users can freely coordinate resources, and the purpose of resource coordination is to maximize the throughput of the whole network, and minimize the performance loss caused by multiplexing. We mainly consider the performance loss of the rate. As shown in Fig. 5, we get the change curve of throughput and performance loss of the whole network caused by the change of the transmit power of the user device under different D2D users. It can be obtained from Fig. 5(a) that as the number of D2D users increases, the throughput gradually increases and is positively correlated with the transmission power of the device. As can be seen from Fig. 5(b), as the number of D2D users increases, the performance loss increases gradually. As the number of user increases, the signal gain of the users to the base station becomes large, and thus the performance loss also becomes large. However, when the number of users is 15 and 20, the performance loss has not changed much. We can get the resource system through CA to improve the throughput of the whole network and reduce the transmission loss.

6. Conclusion

MEC supports distribution of cloud computing functions to the edge of wireless access network, which permits users to execute delay-sensitive and context-sensing applications nearby. At the same time, the pressure of core network can be relieved. On the basis, this paper puts forward emerging model of CA. Every terminal equipment helps to execute resource caching task on MEC and be responsible for the resource collaboration among different device. We propose to use the LSTM to predict the behavior track and application request of terminal equipment. Based on the result of prediction, task caching could be executed on the



(a) The total throughput in D2D



(b) The transmit loss in D2D

Fig. 5. The results of resource collaboration.

edge cloud of neighboring users. We have set the performance evaluation indexes of resource caching, including hit rate and delay of task execution. At the same time, we have set the indexes of network handling capacity and energy consumption aiming at the resource collaboration among different terminals. This is the guide for making scheme in resource collaboration. Finally, simulation experiment is operated to prove that CA can well recognize and help terminal equipment execute resource caching and collaboration and Qos and QoE are improved.

In the future, we need to consider the dynamic nature of communication network and add more uncertain factors for experiment. At the same time, it is necessary to deeply explore the resource allocation and task unloading scheme on MEC. In this way, the resource management system on MEC will have the function of cognition and the network sensing can be strengthened. Last but not least, CA is responsible for storing all the data of users. When collaborating resources, information can be shared with other agents. We will consider to protect users' privacy in the future and guarantee the safety of users' data [55].

Declaration of competing interest

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

Acknowledgment

The authors are grateful to the Deanship of Scientific Research at King Saud University, Saudi Arabia for funding this work through Vice Deanship of Scientific Research Chairs: Chair of Smart Technologies.

References

[1] H. Jiang, C. Cai, X. Ma, Y. Yang, J. Liu, Smart home based on WiFi sensing: A survey, IEEE Access 6 (2018) 13317–13325.
[2] K. Chen, C. Wang, Z. Yin, H. Jiang, G. Tan, Slide: Towards fast and accurate mobile fingerprinting for WiFi indoor positioning systems, IEEE Sens. J. 18 (3) (2018) 1213–1223.
[3] Di Chen, Stephan Schedler, Volker Kuehn, Backhaul traffic balancing and dynamic content-centric clustering for the downlink of fog radio access network, in: 2016 IEEE 17th International Workshop on Signal Processing Advances in Wireless Communications, SPAWC, IEEE, 2016.
[4] M. Chen, Y. Miao, X. Jian, X. Wang, L. Hu, Cognitive-LPWAN: Towards intelligent wireless services in hybrid low power wide area networks, IEEE Trans. Green Commun. Netw. 3 (2) (2019) 407–417.
[5] Kai Lin, Li Chensi, Daxin Tian, Ahmed Ghoneim, M. Shamim Hossain, Syed Umar Amin, Artificial-intelligence-based data analytics for cognitive communication in heterogeneous wireless networks, IEEE Wirel. Commun. 26 (3) (2019) 83–89.
[6] Jing Chen, Kun He, Quan Yuan, Guoliang Xue, Ruiying Du, Lina Wang, Batch identification game model for invalid signatures in wireless mobile networks, IEEE Trans. Mob. Comput. 16 (6) (2017) 1530–1543.
[7] Yin Zhang, et al., Edge intelligence in the cognitive Internet of Things: Improving sensitivity and interactivity, IEEE Netw. 33 (3) (2019) 58–64.
[8] Min Chen, et al., A dynamic service migration mechanism in edge cognitive computing, ACM Trans. Internet Technol. 19 (2) (2019) 30.
[9] Kai Lin, Song Jeungeun, Luo Jiming, Wen Ji, Ahmed Ghoneim, GVT: Green video transmission in the mobile cloud networks, IEEE Trans. Circuits Syst. Video Technol. 27 (1) (2016) 159–169.
[10] J. Zhou, D. Tian, Y. Wang, et al., Reliability-optimal cooperative communication and computing in connected vehicle systems, IEEE Trans. Mob. Comput. (2019).
[11] D. Wang, et al., Adaptive wireless video streaming based on edge computing: Opportunities and approaches, IEEE Trans. Serv. Comput. http://dx.doi.org/10.1109/TSC.2018.2828426.
[12] M. Chen, Y. Hao, H. Gharavi, V. Leung, Cognitive information measurements: A new perspective, Inf. Sci. 505 (2019) 487–497, http://dx.doi.org/10.1016/j.ins.2019.07.046.
[13] M. Chen, V. Leung, From cloud-based communications to cognition-based communications: A computing perspective, Comput. Commun. 128 (2018) 74–79.
[14] Yin Zhang, et al., Emotion-aware multimedia system security, IEEE Trans. Multimed. 21 (3) (2019) 617–624.
[15] Y. Qian, et al., EARS: Emotion-aware recommender system based on hybrid information fusion, Inf. Fusion 46 (2019) 141–146.
[16] M. Chen, Y. Hao, Label-less learning for emotion cognition, IEEE Trans. Neural Netw. Learn. Syst. (2019) http://dx.doi.org/10.1109/TNNLS.2019.2929071.
[17] Y. Zhang, et al., COCME: Content-oriented caching on the mobile edge for wireless communications, IEEE Wirel. Commun. 26 (3) (2019) 26–31.
[18] M. Chen, Y. Hao, C. Lai, D. Wu, Y. Li, K. Hwang, Opportunistic workflow scheduling over Co-located clouds in mobile environment, IEEE Trans. Serv. Comput. 11 (3) (2018) 549–561.
[19] Yixue Hao, et al., Energy efficient task caching and offloading for mobile edge computing, IEEE Access 6 (2018) 11365–11373.
[20] Ke Zhang, et al., Cooperative content caching in 5G networks with mobile edge computing, IEEE Wirel. Commun. 25 (3) (2018) 80–87.
[21] Zhaohui Luo, et al., Energy-efficient caching for mobile edge computing in 5G networks, Appl. Sci. 7 (6) (2017) 557.
[22] Jeroen Famaey, et al., Towards a predictive cache replacement strategy for multimedia content, J. Netw. Comput. Appl. 36 (1) (2013) 219–227.
[23] Zhi Wang, et al., Enhancing internet-scale video service deployment using microblog-based prediction, IEEE Trans. Parallel Distrib. Syst. 26 (3) (2014) 775–785.
[24] Kai Lin, Min Chen, Jing Deng, Mohammad Mehedi Hassan, Giancarlo Fortino, Enhanced fingerprinting and trajectory prediction for IoT localization in smart buildings, IEEE Trans. Autom. Sci. Eng. 13 (3) (2016) 1294–1307.
[25] Yuanyuan Zeng, et al., Smart caching based on user behavior for mobile edge computing, Inform. Sci. (2019).
[26] Jinke Ren, et al., Collaborative cloud and edge computing for latency minimization, IEEE Trans. Veh. Technol. 68 (5) (2019) 5031–5044.
[27] Tuyen X. Tran, et al., Collaborative mobile edge computing in 5G networks: New paradigms, scenarios, and challenges. arXiv preprint arXiv:1612.03184, 2016.
[28] M. Chen, W. Li, Y. Hao, Y. Qian, I. Humar, Edge cognitive computing based smart healthcare system, Future Gener. Comput. Syst. 86 (2018) 403–411.
[29] Kai Lin, Jiming Luo, Long Hu, M. Shamim Hossain, Ahmed Ghoneim, Localization based on social big data analysis in the vehicular networks, IEEE Trans. Ind. Inf. 13 (4) (2017) 1932–1940.
[30] M. Chen, Jun Yang, X. Zhu, X. Wang, M. Liu, J. Song, Smart home 2.0: innovative smart home system powered by botanical IoT and emotion detection, Mobile Netw. Appl. 22 (2017) 1159–1169.
[31] Yin Zhang, et al., SOVCAN: Safety-oriented vehicular controller area network, IEEE Commun. 55 (8) (2017) 94–99.
[32] Kun He, Jing Chen, Ruiying Du, Qianhong Wu, Guoliang Xue, Xiang Zhang, DeyPoS: Deduplicatable dynamic proof of storage for multi-user environments, IEEE Trans. Comput. 65 (12) (2016) 3631–3645.
[33] Min Chen, et al., Mobile agent-based directed diffusion in wireless sensor networks, EURASIP J. Adv. Signal Process. 2007 (1) (2007) 036871.
[34] Damianos Gavalas, et al., Mobile agent itinerary planning for WSN data fusion: considering multiple sinks and heterogeneous networks, Int. J. Commun. Syst. 30 (8) (2017) e3184.
[35] Kai Daniel, et al., Cognitive agent mobility for aerial sensor networks, IEEE Sens. J. 11 (11) (2011) 2671–2682.
[36] Hridya C. Premkumar, Vijayashree R. Budyal, Mahabaleshwar S. Kakkasageri, Cognitive agent based stable routing protocol for vehicle-to-vehicle communication, in: 2016 IEEE Annual India Conference, INDICON, IEEE, 2016.
[37] Ho Florence, et al., Multi-agent path finding for UAV traffic management, in: Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems, International Foundation for Autonomous Agents and Multiagent Systems, 2019.
[38] Jingjing Cui, Liu Yuanwei, Nallanathan Arumugam, Multi-agent reinforcement learning based resource allocation for UAV networks. arXiv preprint arXiv:1810.10408, 2018.
[39] Xiaofei Wang, et al., In-edge ai: Intelligentizing mobile edge computing, caching and communication by federated learning. arXiv preprint arXiv:1809.078572018.
[40] Jing Chen, Chiheng Wang, Ziming Zhao, Kai Chen, Ruiying Du, Gail-Joon Ahn, Uncovering the face of Android ransomware: Characterization and real-time detection, IEEE Trans. Inf. Forensics Secur. 13 (5) (2018) 1286–1300.
[41] Min Chen, et al., Data-driven computing and caching in 5G networks: Architecture and delay analysis, IEEE Wirel. Commun. 25 (1) (2018) 70–75.
[42] Abdallah Khreishah, Jacob Chakareski, Collaborative caching for multicell-coordinated systems, in: 2015 IEEE Conference on Computer Communications Workshops, INFOCOM WKSHPS, IEEE, 2015.
[43] Zhi Wang, et al., Enhancing internet-scale video service deployment using microblog-based prediction, IEEE Trans. Parallel Distrib. Syst. 26 (3) (2014) 775–785.

[44] Suoheng Li, et al., Popularity-driven content caching, in: IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications, IEEE, 2016.
[45] Sabrina Müller, et al., Context-aware proactive content caching with service differentiation in wireless networks, IEEE Trans. Wireless Commun. 16 (2) (2016) 1024–1036.
[46] Fei Shen, et al., A stackelberg game for incentive proactive caching mechanisms in wireless networks, in: 2016 IEEE Global Communications Conference, GLOBECOM, IEEE, 2016.
[47] Hye J. Kang, et al., Mobile caching policies for device-to-device (D2D) content delivery networking, in: 2014 IEEE conference on computer communications workshops, INFOCOM WKSHPS, IEEE, 2014.
[48] S.M. Shahrear Tanzil, William Hoiles, Vikram Krishnamurthy, Adaptive scheme for caching youtube content in a cellular network: Machine learning approach, IEEE Access 5 (2017) 5870–5881.
[49] Min Chen, et al., Edge-CoCaCo: Toward joint optimization of computation, caching, and communication on edge cloud, IEEE Wirel. Commun. 25 (3) (2018) ppn1–27.
[50] Yaping Sun, et al., Communications, caching and computing for mobile virtual reality: Modeling and tradeoff, IEEE Trans. Commun. (2019).
[51] Anselme Ndikumana, et al., Joint communication, computation, caching, and control in big data multi-access edge computing, IEEE Trans. Mob. Comput. (2018).
[52] Daxin Tian, et al., Channel access optimization with adaptive congestion pricing for cognitive vehicular networks: An evolutionary game approach, IEEE Trans. Mob. Comput. (2019) 1.
[53] Min Chen, et al., Cognitive internet of vehicles, Comput. Commun. 120 (2018) 58–70.
[54] Daxin Tian, et al., Self-organized relay selection for cooperative transmission in vehicular Ad-hoc networks, IEEE Trans. Veh. Technol. (2017) 1.
[55] P. Zhao, J. Li, F. Zeng, F. Xiao, C. Wang, H. Jiang, ILLIA: Enabling k-anonymity-based privacy preserving against location injection attacks in continuous LBS queries, IEEE Internet Things J. 5 (2) (2018) 1033–1042.

Rui Wang is currently a master student at Embedded and Pervasive Computing (EPIC) Laboratory in the School of Computer Science and Technology, HUST. She received her bachelor degree of Computer Science and Technology from College of Information Science and Engineering of Lanzhou University in 2018, China. Her research interest is focused on cognitive computing and big data analysis etc.

Miao Li is currently pursuing a PhD in the Embedded and Pervasive Computing (EPIC) Laboratory in the School of Computer Science and Technology, HUST. She received her bachelor's degree in applied mathematics from the school of mathematics and statistics, zhengzhou university in 2016. Her research interests are mainly in machine learning, cognitive computing and edge computing.

Limei Peng is an assistant professor in the School of Computer and Engineering, Kyungpook National University, South Korea, since 2018. She has been an Assistant Professor in the Department of Industrial Engineering, Ajou University, South Korea, from 2014 to 2018. She has authored/co-authored for more than 90 research papers in the leading journals and conferences. She has been a guest editor in several journals, such as IEEE ACCESS, Elsevier journal of optical switching and networking, Springer journal of Mobile Networks and Applications, etc. Her research interests include the Cloud computing, Edge computing, data communications, optical communication networks and protocols, cellular networks, 5G, etc.

Ying Hu is now a lecturer in School of Mathematics and Economics, Hubei University of Education. He worked as a Poster-Doctoral Fellow in Embedded and Pervasive Computing Lab (EPIC) at Huazhong University of Science and Technology (HUST) from Sep. 2014 to Jan. 2017. He received his B. S, M.S and Ph.D from Central China Normal University (CCNU), Harbin Institute of Technology (HIT) and Dalian University of Technology (DUT) in 2003, 2008 and 2014. His recent research interests include big data in healthcare, online social information diffusion and medical image analysis, etc.

Mohammad Mehedi Hassan (M'12) is currently an Associate Professor of Information Systems Department in the College of Computer and Information Sciences (CCIS), King Saud University (KSU), Riyadh, Kingdom of Saudi Arabia. He received his Ph.D. degree in Computer Engineering from Kyung Hee University, South Korea in February 2011. He received Best Journal Paper Award from IEEE Systems Journal in 2018. He also received Best Paper Award from CloudComp conference at China in 2014. He received Excellence in Research Award from CCIS, KSU in 2015 and 2016 respectively. He has published over 130+ research papers in the journals and conferences of international repute. He has also played role of the guest editor of several international ISI-indexed journals. His research areas of interest are cloud federation, multimedia cloud, sensor-cloud, Internet of things, Big data, mobile cloud, sensor network, publish/subscribe system and recommender system. He is a senior member of IEEE.

Abdulhameed Alelaiwi is currently an Associate Professor of Software Engineering Department in the College of Computer and Information Sciences (CCIS), King Saud University (KSU), Riyadh, Kingdom of Saudi Arabia. He received his PhD degree in Software Engineering from the College of Engineering, Florida Institute of Technology-Melbourne, USA in 2002. He is currently serving as the Vice Dean of Research Chairs Program at KSU. He has authored and co-authored many publications. He has published over 70+ research papers in the ISI-Indexed journals of international repute. He has served as Technical Program Committee member in numerous reputed international conferences/workshops. His research interest includes software testing analysis and design, cloud computing, multimedia, Internet of things, Big data, and mobile cloud. He is member of IEEE.