Cognitive Popularity based AI Service Sharing for Software-Defined Information-Centric Networks

Siyi Liao, Jun Wu, Jianhua Li, Ali Kashif Bashir, Shahid Mumtaz, Alireza Jolfaei, and Nida Kvedaraite

***Abstract*—As an important architecture of next-generation network, Software-Defined Information-Centric Networking (SD-ICN) enables flexible and fast content sharing in beyond the fifth-generation (B5G). The clear advantages of SD-ICN in fast and efficient content distribution and flexible control make it a perfect platform for solving the rapid sharing and cognitive caching of AI services, including data samples sharing and pre-trained models transferring. With the explosive growth of decentralized artificial intelligence (AI) services, the training and sharing efficiency of edge AI is affected. Various applications usually request the same AI samples and training models, but the efficient and cognitive sharing of AI services remain unsolved. To address these issues, we propose a cognitive popularity-based AI service distribution architecture based on SD-ICN. First, an SD-ICN enabled edge training scheme is proposed to generate accurate AI service models over decentralized big data samples. Second, Pure Birth Process (PBP) and error correction-based AI service caching and distribution schemes are proposed, which provides user request-oriented cognitive popularity model for caching and distribution optimization. Simulation results indicate the superiority of the proposed architecture, and the proposed cognitive SD-ICN scheme has 62.11% improved to the conventional methods.**

***Index Terms*—cognitive popularity, decentralized big data, Software Defined Information-Centric Network (SD-ICN), service sharing.**

I. Introduction

RECENTLY, Software-Defined Networking (SDN) and Information-Centric Networking (ICN) have been extensively studied as the mainstream architectures of the next-generation network. When SDN meets ICN, they will greatly enhance network management, such as traffic engineering, routing and service chaining [?]. The separation of control and forwarding of SDN and the well-developed OpenFlow protocol can be combined with the characteristics of dynamic naming and efficient content distribution of ICN. Therefore, as an integration of them, Software-Defined Information-Centric Networking (SD-ICN) has become an important content sharing and distributing network architecture in the beyond fifth-generation (B5G) [?]. It offers centralized control and in-network caching which makes it more ideal for a wide range of network devices. In the in-network caching and content distribution scheme of SD-ICN, every intermediate SD-ICN switch can provide interest or data on behalf of the original producer, reducing the so-called flash crowd situation [?]. Recent studies pointed out that content sharing service scales better in an SD-ICN architecture than the traditional host-centric IP model [?]. The clear advantages of SD-ICN in fast and efficient data transmission, content distribution and reliability assurance make it a very promising network model for AI service sharing, including data samples sharing and pre-trained models transferring [?].

Recognizing, discovering and extracting potential patterns from massive data is the core utility of big data analytics as it results in higher levels of insights for decision making and trend prediction [?]. Therefore, massive data capturing devices and sensors are deployed and distributed to gather continuous data for edge learning applications. Thanks to the recent advancement in fast computing, efficient storage and novel machine learning algorithms, more attention has been drawn in the area of big data analytics and knowledge extraction for diverse applications. Fast and efficient connectivity of these smart devices enables many valuable and remarkable applications like smart home, intelligent transport, e-health, smart grid, and smart cities [?]. Decentralized big data and machine learning tasks are fully integrated, bringing intelligence and cognition to the network.

However, the widespread deployment of edge learning also brings new problems that are not met by existing models (ie. homogenous learning models are repeatedly trained, models with smaller data volumes are easily over-fitting, etc.). The users will inevitably produce similar machine learning tasks, they need the same type of data, and even expect the same training results. For example, with the development of transfer learning, some models must be further trained on the existing initial model. This will also place higher demands on the sharing and distribution of the AI model. Without an effective sharing and fast distribution scheme, a large number of tasks will be repeatedly trained at the IoT edge and the same type of data cannot be aggregated and used as well. Whats more, a smaller amount of data samples will also lead to over-fitting of the model. Therefore, invalid and meaningless model training will be widespread, resulting in a waste of resources and a decline in Quality of Service (QoS). How to integrate decentralized data of the network for the training of AI models

This work was supported in part by the National Natural Science Foundation of China under Grant 61972255.(*Corresponding author: Jun Wu.*)

Jianan Li, Jun Wu and Siyi Liao are with Shanghai Key Laboratory of Integrated Administration Technologies for Information Security, School of Cyber Security, Shanghai Jiaotong University, Shanghai 200240, China.(e-mail: junwuhn@sjtu.edu.cn)

Ali Kashif Bashir is with Department of Computing and Mathematics, Manchester Metropolitan University, UK, and School of Electrical Engineering and Computer Science, National University of Science and Technology, Islamabad (NUST), Islamabad, Pakistan.

Shahid Mumtaz is with Instituto de Telecomunicaes (IT), Portugal.

Alireza Jolfaei is with Department of Computing, Macquarie University, Sydney NSW 2109, Australia.

Nida Kvedaraite is with Kaunas University of Technology (KTU), Republic of Lithuania.

remains to be resolved. Therefore, both the decentralized big data resources and the machine learning models are in need of effective management and cognitive sharing scheme.

To address these issues, We make full use of the advantages of SD-ICN to enable the network devices with the ability to cognize the popularity of the AI service. The contributions of our work are summarized as follows.

- An SD-ICN based architecture that enables the efficient sharing and fast distribution of AI service is proposed, including data samples sharing and pre-trained models transferring. We have hierarchically defined the functions and relationships between the various functional models. Different packets in the proposed architecture are illustrated in detail.
- A novel scheme for effective edge training with decentralized big data is proposed. In order to solve the problem of repeat training and decline in model accuracy, similar AI services with the same type of samples and training results are aggregated and trained in the proposed SD-ICN architecture by using the decentralized big data.
- We propose a cognitive popularity model for the optimization of SD-ICN caching and distributing. We present a detailed mathematical model to optimize the cache space of the SD-ICN nodes and increase cache hit ratio based on Pure Birth Process (PBP) and error correction. Through the prediction and ranking of the request, we implemented a dynamic update of the cache.

The remainder of this paper is organized as follows. The related work is given In section II and the strengths of the proposed scheme are described. Our system model of scheme is presented in Section III. Both the basic implementation of the architecture and design principles analysis in detail are provided in Section IV. Simulation results are shown in Section V to estimate the performance of the scheme. Final conclusions are drawn in Section VI.

II. Related Work

Related methods, including blockchain, AI and have been extensively studied so as to provide the network with faster and safer services [?] [?]. The development and expansion of IoT makes it one of the major sources of big data, as it connect a myrid of sensors and smart devices together to share their captured status of the environments [?]. This also enables the huge potential of edge AI and its related technologies, including applications in transportation, medical, and security [?] [?]. Voluminous amounts of data have been produced and used, since the past decade as the miniaturization and universalization of IoT devices [?]. Authors of [?] proposed a heuristic approach in the edge-cloud-hybrid system for IoT so as to increase the efficiency of big data service deployment. By the monitoring of multiple factors, an innovative system is presented for the detection and support of Obtrusive Sleep Apnea (OSA) of elderly people using the available open data [?]. A novel price forecasting model for Smart Grid is introduced in [?] by the integration of Differential Evolution (DE) and Support Vector Machine (SVM) classifier. Authors of [?] presented and discussed a scalable and flexible Deep Learning (DL) framework based on Apache Spark for mobile big data, which enables the orchestration of DL models with a large number of hidden layers and parameters on a computing cluster.

On the other hand, the architecture of the next-generation network has been widely discussed [?]. Based on the Software Defined Network (SDN) technology, authors of [?] proposed a content popularity prediction based on deep learning to achieve the popularity prediction. It uses the computing resource and link of SDN to build a distributed and reconfigurable deep learning network. As an content-centric approach, ICN have been recently regarded as an alternative to the traditional host-centric network paradigm [?]. Obvious benefits of ICN in terms of improved interest/content sharing scheme and better reliability has already raised ICN as highly promising networking techeology for environments such as IoT [?]. A novel cognitive ocean network (CONet) architecture is proposed as well as its important and useful demonstration applications [?]. Authors of [?] proposed an ICN-IoT architecture in which ICN nodes provide IoT gateways capabilities and ICN in-network caching. In order to improve the energy efficiency of IoT, the in-network caching of ICN is leveraged by authors of [?] to propose a novel cooperative caching scheme based on the IoT data lifetime and user request rate. Based on NDN, MR-IoT defines schemes to execute MapReduce tasks on IoT including computational tree construction and computational task dissemination [?]. With IoT and ICN combined all together with the Edge Computing concept, the cability of merging DL models is discussed and studied in [?], such as Convolutional Neural Network (CNN), Recurrent Neural Network (RNN), and Reinforcement Learning (RL). Although the research on ICN and IoT has been extensive, few works have focus on applying ICN to the sharing of AI services and data locations in IoT. Efficient decentralized big data-based AI services are in urgent need.

III. Basic Architecture

A. Proposed Artificial Intelligence Service Sharing

AI-capable devices are widely deployed at the edge of the network and are deeply integrated with smart cities. The aggregation of massive decentralized data and the richness of edge computing resources at the edge of the network have stimulated a wide variety of artificial intelligence-based services and applications. However, in heterogeneous IoT networks, computing resources and edge data are often unbalanced, and it is almost impossible to train models based on every demand of user service. On the other hand, the extensive similar service has been repeatedly requested by various applications and users.

As shown in Fig.??, we regard the fog server as the basic unit of content caching and distribution, and form an information-centric network that connects massive IoT devices and enable the sharing of various AI-based services. In the proposed scheme, pre-trained Machine Learning (ML) models and decentralized big data are cached in the fog node. When a user requests a certain type of AI service, it first queries whether the service model has been cached in the local fog node. If

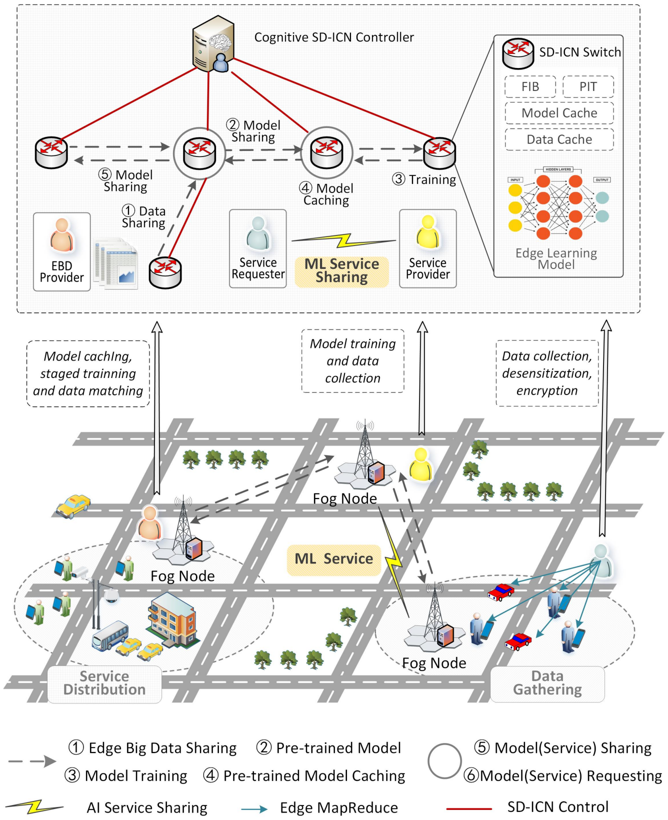


Fig. 1: Scenarios of AI service sharing using SD-ICN.

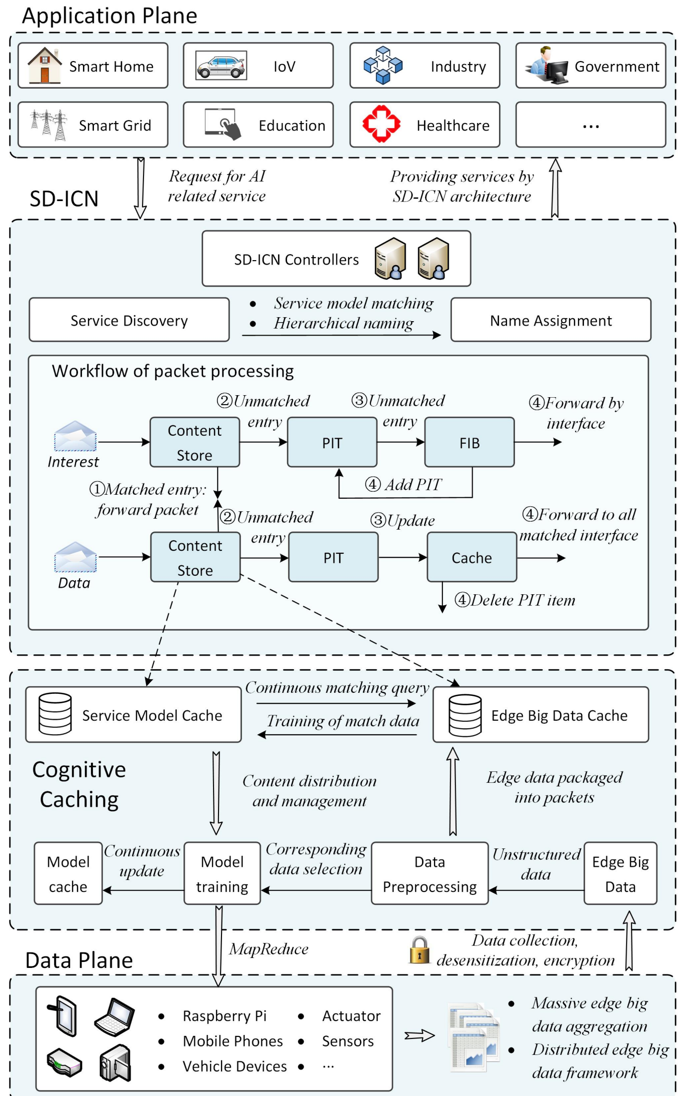


Fig. 2: Architecture of integrated decentralized big data and SD-ICN for AI service sharing.

the model is not available locally, the fog node broadcasts an interest packet to other nodes in the network, waiting for the returned information. If there is no ML model required by the service in the network, the local fog node determines whether to request the same type of data from other nodes according to the service precision requirement and the local data amount. If there is a large amount of relevant data locally, the fog node calculates the model locally or requests nodes with strong computing power nearby to complete the computing task. When there is not enough data locally to support the model training, the fog node broadcasts an interest packet and waiting for feedback from other nodes. The fog server therefore searches for workers as an edge master for AI model training in an Edge MapReduce (EMR) approach so as to make full use of computing resources at the edge of the network and achieve rapid model training. EMR implements reliability by distributing training of AI services to each available agent in Fog, and the agents periodically return the latest states of model training. With the advantage of SD-ICN, decentralized big data based artificial intelligence services can be efficiently and quickly distributed and cached at the edge of the network, alleviating the imbalance of computing resources and edge data. With the SD-ICN, the application of decentralized big data applications and AI service management is enhanced, providing a natural platform for edge intelligence.

B. Basic Architecture of the SD-ICN for AI Sharing

Because of the advantages of ML in forecasting, decision making, perception and recognition, ML-based artificial intelligence services have penetrated into all areas of smart cities, including Industrial Internet of Things (IIoT), e-health, smart home, Intelligent Transportation Systems (ITS) and education. Due to the large scale of the aforementioned application scenarios, the same type of service is often requested by geographically distributed users. However, the existing traditional IP-centric networks are not efficient enough for high-speed content distribution [**?**]. When a certain user needs to continuously request a large amount of data, the traditional network is not sensitive to the specific content of the request. This has been significantly improved in the content-centric SD-ICN architecture. Frequent task requests and efficient content distribution need to be implemented by the caching and forwarding mechanism of SD-ICN.

In the proposed architecture shown in Fig.**??**, SD-ICN directly interfaces with the needs of upper-level users. For the upper Service Layer, ML-based models and edge data are hierarchically named according to the service requests of users and applications. The name act as the vital part of interest packets and data packets in order to request relevant content from the SD-ICN. All nodes in the SD-ICN network follow the same set of naming and matching rules. The model that needs the data matches the corresponding packet name in the fog node, so the same type of data packet can be used for training of multiple models.

The SD-ICN Layer is the core for the implementation of AI sharing and distributing. When the fog node receives an interest packet of a certain kind of data or service, it first queries whether there is a corresponding content that can be matched in the local cache to be returned. If there is no matching local-cached content in the cache, the Pending Interest Table (PIT) is queried, and if it exists in the PIT, the PIT is updated and the packet is discarded. If the interest package PIT is not matched, continue to query the Forwarding Information Base (FIB). If the request can be matched in the FIB, the interest packet is forwarded to the destination node according to the defined f orwarding r ule a nd t he interface, and the content of the PIT is updated. Otherwise, the interest packet is discarded. On the other hand, when the fog node receives the data packet, it also first q ueries t he l ocal cache. The packet is dropped if the packet already exists in the cache. Otherwise, continue to query the PIT. If the packet is matched in the PIT, the packet is cached and forwarded to all matching interfaces in the PIT, and the entry is subsequently removed from the PIT. If no match is found in the PIT, which means the data packet is irrelevant to the service or potential service of the node, and the data packet is discarded.

The Cognitive Caching ayer mainly implements model training and popularity-based cache. Massive edge data is aggregated in the fog node and preprocessed, structured, and secured. In addition to the current edge functions such as edge computing and device management, the fog nodes mainly act as the platform to manage the cached content and pre-processing of the data in the proposed architecture. The fog node can pre-process the data, perceive the local potential service, and name the data in the same hierarchical way as the service layer. For locally generated ML tasks, the fog node provides services to local users in the order of: 1) local model cache 2) broadcast model request 3) local data training 4) request data for training.

C. Packet Definition in Proposed Architecture

The traditional ICN has only two kinds of packets, namely, interest packets and data packets. In order to adapt to the AI service sharing model proposed in this paper, this paper further subdivides the function of the data packet. Fig.**??** shows the four kinds of packets in the proposed architecture: Service Interest Packet (SIP), Data Interest Packet (DIP), Service Model Packet (SMP) and Data Packet (DP). We have defined the meaning of the fields in the packets as shown in Fig.**??**. Different from the traditional ICN, the cache space is divided into a data cache and a service model cache. They optimize

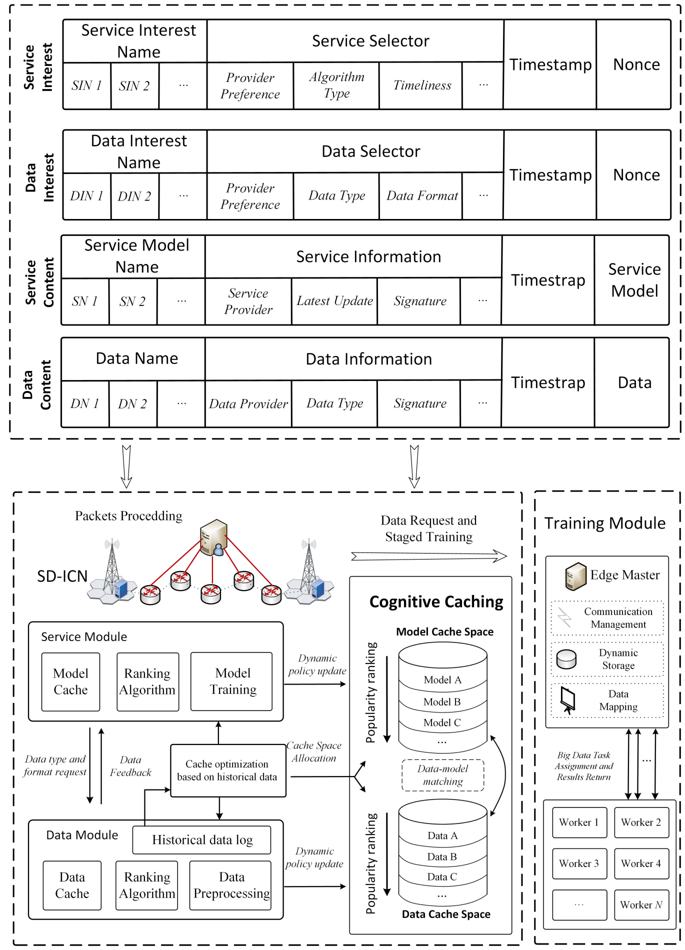


Fig. 3: Training, caching and distribution for AI service sharing by cognitive SD-ICN.

the cache based on the content popularity to ensure a higher cache hit rate. The fog node records the historical data of the request and response by the access log, and uses this as a basis to optimize the limited cache space. Each node optimizes its own cache based on its calculated popularity, to ensure that higher-popular content is more likely to be returned.

The specific functions of each type of data packet are as follows:

1) Service Interest Packet (SIP): SIP is used to issue a request for a specific AI model. SIP includes requirements for service provider preferences, algorithm types, and the most recent update time of the model.

2) Data Interest Packet (DIP): DIP is used to request a certain type of data for training the model. DIP mainly emphasizes various aspects of the requested data, including data type, format, provider, security, etc.

3) Service Model Packet (SMP): The function of SMP is to return the pre-trained AI model. Different aspects of the information with the AI service are included in the header, including the source provider of the model, the updater, the update time, and the corresponding data type.

4) Data Packet (DP): DP returns the specified type of data after preprocessing. The information in the DP header is mainly used to match the requirements of the DIP.

D. Decentralized Big Data Management and Training

As the Edge Master (EM), the fog server distributes big data tasks to Edge Workers (EW) in a MapReduce way, thereby making full use of the idle resources at the IoT edge. Based on the computing power of the EWs, the EM dynamically allocates data and computational tasks to the EWs and summarizes the parameters after the calculation is complete. It is worthwhile to consider that, even for models that have already been cached, further improvement in model accuracy is still needed. Therefore, the data-model matching and the staged training of the pre-trained models proposed below are also deployed in the fog node. Therefore, we have established an interaction mechanism for the cached decentralized big data and the AI model. In general, the performance of the ML model will improve as the amount of training data increases. So the model cached in SD-ICN should also be improved by corresponding data continuously, which is called Staged Training. As shown in Fig.**??**, each fog node has a model cache space and a data cache space, respectively. The pre-trained models and the data generated and acquired by the fog node are cached here. Through the model-data matching mechanism, packets are matched with the names in the model cache before being discarded or after being cached. When the cache model of a fog node matches the data, the existing model will be further trained with local data to make the model. In this way, the accuracy of the model will be improved in stages. On the other hand, before the node drops a useless data packet, it also matches the packet with the existing model cache for the update of models.

IV. Problem Formulation and Proposed Algorithm

A. Problem Formulation of Cognitive Cache Popularity

Edge fog nodes have limited cache space and can only store a limited number of service models. Therefore, we need to reasonably allocate the cache of the fog node based on the historical request data to provide timely service for more likely service requests. So we can think of the edge cache problem as a problem based on historical request and service data to determine the allocation of cache space at the current time or in the future.

We assume that a service model is cached in the fog node at time T_0 for service sharing. At the current time T, the total service time of the model is $t = T - T_0$. For the service request model of the fog node, we first consider the proportion of different types of service requests. Taking into account the time-varying data throughput, communication resources and requests of each node, it is inevitable that the number of requests as a measure of popularity will bring some errors. So we define the following Request Index (RI).

Definition 1: We define the RI of service i as the ratio of the request number of $i - th$ request to the total n service requests in the past period of time.

$$R_i(t) = \frac{\sum_{t\in\Delta t} r_{i,t}}{\sum_{t\in\Delta t}\sum_{j=1}^{n} r_{j,t}\cdot\delta_{j,t}} \tag{1}$$

TABLE I: Main symbols and explanations

Symbols	***Explains***
T	Current time
$t = T - T_0$	the total service time
RI	Request Index
$R_i(t)$	The ratio of the $i - th$ service request number to the total n requests
$N(ti)$	Number of requests at time t_i
$P(\cdot)$	The probability of a certain state
$E[\cdot]$	The expectation of a certain state
λ	The birth rate of PBP
$r(t_i)$	A set of discrete RI values
X	The increment from $r(t_{i-1})$ to $r(t_i)$
$\Psi(t_i)$	equivalent birth rate contribution of RI
$\overline{R}$	Expectation of cumulative RI
RPE	Difference between the expected RI and RI
RPC	Corrected error value
$R_{c,x}$	final request index of service x

A popularity-based cache optimization method can be obtained by predicting the RI of all service requests received by the fog node.

B. Pure Birth Process of Cache Popularity

The Pure Birth Process (PBP) is a special Markov Chain (MC) that is widely used in growth prediction, data analysis, and fault replacement [**?**] [**?**]. The core idea of cache optimization is to make a reasonable allocation of future caches based on historical request data, which is consistent with the basic idea of PBP. Therefore, we can treat the historical request data of a service received by the fog node as a PBP and predict the request for a certain period of time, thereby optimizing the fog node cache space. A non-decreasing, non-negative, integer form of continuous-time Markovian process can be called a PBP if it satisfies the following equation. For the PBP in the fog node cache optimization, it can be denote as $\{N(t_i) \mid i \in [0, n]\}$ and we use the following equations to represent the process:

$$\begin{cases} P\left\{N(t_{i+1}) - N(t_i) = 1 \mid N(t_i = k)\right\} \\ \quad \lambda_k(t_{i+1} - t_i) + ok(t_{i+1} - t_i) \\ \\ P\left\{N(t_{i+1}) - N(ti) = 0 \mid N(t_i = k)\right\} \\ \quad 1 - \lambda_k(t_{i+1} - t_i) + ok(t_{i+1} - t_i) \\ \\ \quad N(0) = 0 \end{cases} \tag{2}$$

where $P(\cdot)$ represents the probability of a certain state, $P\left\{N(t_{i+1}) - N(ti) = 0 \mid N(t_i = k)\right\}$ in the above equation represents the probability of $N(ti)$ growth and λ_k is the birth rate of PBP. $N(0) = 0$ is the initial condition of the caching process, indicating that there is no cache of the content in the fog node at the beginning. Through the initial conditions of PBP we can get the following difference equations for this process.

$$\begin{cases} \frac{dP_0(t_i)}{dt} = P_0'(t_i) = -\lambda_k \cdot P_0(t_i) \\ \frac{dP_k(t_i)}{dt} = P_m'(t_i) = -\lambda_k \cdot P_k(t_i) + \lambda_{k-1} \cdot P_{k-1}(t_i) \end{cases} \quad (3)$$

Eq.(3) is the differentiation of Eq.(2) to t. Solve the above difference equation, we can get:

$$P_k(t_i) = \lambda_{k-1} e^{-\lambda_k ti} \int_0^{ti} e^{\lambda_m x} P_{k-1}(x) dx \quad (4)$$

It is worth noting that although we get a general solution to the growth probability here, this is not for the final cache optimization prediction. λ_k is the birth rate parameter in PBP. Since λ_k is unknown, we need to further calculate it based on the historical data of the cache request. In addition, since Eq. (1) is calculated not by an integer but by a percentage. So we keep the result of $R_i(t)$ with a 4-digit significant digit and multiply it by 10^4. Therefore, $R_i(t)$ can be used as an integer in the PBP.

C. Popularity Prediction Based Cache Optimization

To solve the above problem and combine our popularity prediction with PBP, we abstract the popularity prediction problem into a linear PBP. The empirical distribution of the RI indicates that the contribution of the historical cache content growth rate λ to the current time prediction is not constant, but changes over time. The contribution of our birth rate to popularity is regarded as a function of the total service time t of the model, making this process a linear PBP. We describe this process as follows.

A series $\{r(t_i) \mid i \in [0, n]\}$ is a set of discrete RI values that vary with time as measured by the fog node. This series is considered to be a linear PBP, and the birth rate $\lambda_{r(t_i)}$ can be expressed as $\lambda_{r(t_i)} = r(t_i)\psi(t_i)$. With Eq.(3), Eq.(4) and $\lambda_{r(t_i)} = r(t_i)\psi(t_i)$, the differential equation of Liner PBP can be rewritten into the following form.

$$\begin{cases} dP_{r(t_{i-1})}(t_i)/dt = -r(t_{i-1})\psi(t)P_{r(t_{i-1})}(t_i) \\ dP_{r(t_{i-1}+X)}(t_i)/dt = P'_{r(t_{i-1}+X)}(t_i) = \\ \quad -(r(t_{i-1}+X)\psi(t_i)P_{r(t_{i-1}+X)}(t_i) \\ \quad +(r(t_{i-1}+X-1)\psi(t_i)P_{r(t_{i-1}+X-1)}(t_i) \end{cases} \quad (5)$$

where X is the increment from $r(t_{i-1})$ to $r(t_i)$ and $\psi(t_i)$ is the contribution of historical RI to the birth rate at time t_i. By the general form Eq. (5) of the solution of the difference equation, the above differential equation has the following solution.

$$P_{r(t_i)}(t_i) = P_{r(t_{i-1}+X)}(t_i) = \binom{r(t_{i-1}) + X - 1}{X} \cdot e^{-r(t_{i-1})\psi(t_i)(t_i - t_{i-1})}(1 - e^{-\psi(t_i)(t_i - t_{i-1})})^X \quad (6)$$

Through the above solution, we can get the expected situation of the request index of the specific content over a period of time, that is, the expression of the probability distribution $P_{r(t_i)}(t_i)$ of the request index RI within the period of time t_{i-1} to t_i. By observing Eq.(7) we find that $P_{r(t_i)}(t_i)$ satisfies the conditions of the Pascal distribution. The Pascal distribution is a positive integer form of the negative binomial distribution, which describes the probability that the nth success occurs at the xth time and is a statistically discrete probability distribution. For this Pascal distribution $P_{r(t_i)}(t_i)$, we can get its conditional expectation $E[r(t_i) \mid r(t_{i-1})]$ as follows.

$$\begin{aligned} \overline{R_{t_i,t_{i-1}}} &= E[r(t_i) \mid r(t_{i-1})] \\ &= r(t_{i-1}) \cdot e^{\psi(t_i)(t_i - t_{i-1})} \end{aligned} \quad (7)$$

Through Eq.(8) we can get the popularity expectation $\overline{R_{t_i,t_{i-1}}}$ at time t. According to the initial conditions of the cache and the conditional probability, we can get the cumulative request index value $\overline{R} = E[r(t_i)]$ at time t.

$$\overline{R} = E[r(t_i)] = r(t_i) \cdot e^{\Psi(t_i)(t_i - t_{i-1})} \quad (8)$$

where $\Psi(t_i)$ is used to represent the equivalent birth rate contribution of the cumulative request index RI requested at time t_i. We finally need to get the value of $E[X] = E[r(t_i) - r(t_{i-1})]$, which is the expected request index RI from time t_i to t_{i-1}.

$$\begin{aligned} &E[X] = E[r(t_i) - r(t_{i-1})] = \\ &r(t_{i-1}) \cdot [e^{\Psi(t_i)}/e^{\Psi(t_{i-1})} - 1] \end{aligned} \quad (9)$$

Next, we only need to solve for $\Psi(t_i)$ to get the RI we need by the historic service request data.

The request index at a given service time t is proportional to its probability at that time. In statistics, the probability $P(t_i)$ may be the ratio of the time request index to the total request index. So we can calculate $P(t_i)$ in the following way

$$P(t_i) = \lim_{T \to \infty} E_T[r(t_i)] \cdot \alpha^{-t_i} / \sum_{k=1}^{t_{max}} E_T[r_k \cdot \alpha_{-k}] \quad (10)$$

where E_T is the request index at of service time t_i at time T and α is the discount factor in time dimension. t_{max} stands for the recorded longest service time. $E_T[r(t_i)]$ can be express as:

$$E_T[r(t_i)] = [\sum_{k \mid k \in \Omega_{t_i}^T} r_k(t_i)/n_{t_i}^T] \quad (11)$$

where Ω_{t_i} denotes the request set of service time t_i at time T and n is the number of total service request. From the above expressions, it is not difficult to see that $P(t_i)$ and $E[r(t_i)]\alpha^{-t_i}$ are clearly proportional to each other. So we can write other items as constant C. Therefore, the sum of request index for a service can be expressed as the following form in a statistical perspective.

$$E[r(t_i)] = r(t_r) + C \sum_{k=t_r}^{t_i} \alpha^j \cdot P(k) \quad (12)$$

Algorithm 1 Cache Optimization Algorithm based on Pure Birth Process

Input: Historical data set of request, time t to be predicted, size of cache space k

Output: Cache strategy

Count the total number N of request types
Calculate the request index $R_i(t)$ for all requests
Determine the time factor α
Count the total number of request items c
  if $c > k$ **do**
    for each kind of request R_i $i = 0, \cdots, N$ **do**
      Establish a PBP differential equation
      Find the solution $P_{r(t_i)}(t_i)$ of the Eq.(6)
      Calculate expect cumulative RI $\overline{R} = E[r(t_i)]$
      Calculate $E[r(t_i)]$ and $\Psi(t_i)$by data statistics
      Calculate the estimated value of RI by Eq.(10)
      Error correction by Eq.(16)-Eq.(18)
    end
  for $i = 0, \cdots, N$ **do**
    Sort by predicted $R_{c,x}$ value
    Select the first k items as the cache for the next moment
  else
    Cache all the selected content
Update cache policy

We make Eq.(9) equal to Eq.(13) and get the following equation. That is, the expectation of the cumulative RI value obtained from the statistical point of view is equal to the expected RI value from the perspective of the random process.

$$r(t_i) \cdot e^{\Psi(t_i)(t_i - t_{i-1})} = r(t_r) + C \sum_{k=t_r}^{t_i} \alpha^j \cdot P(k) \tag{13}$$

We can solve $\Psi(t_i)$ from the above equation as long as we set a appropriate $alpha$. Therefore, $\Psi(t_i)$ can be rewritten as:

$$\Psi(t_i) = ln(C \sum_{k=t_r}^{t_i} \alpha^j \cdot P(j)/r(t_r+1))/(t_i - t_{i-1}) \tag{14}$$

Definition 2: Request Prediction Error (RPE) refers to the difference between the expected request index obtained by applying the above PBP-based method and the real request index, which can be expressed by Eq.(16).

$$RPE = \frac{\overline{R_{x,t_i}}}{\sum_{j=1}^{n} \overline{R_{j,t_i}} \cdot \xi_{j,t_i}} - \frac{R_{x,t_i}}{\sum_{k=1}^{m} R_{k,t_i}} \tag{15}$$

However, in the prediction process, there is a certain degree of error in the predicted value due to the influence of the initial value and the predicted time. In this paper, the correction value of the service request error is calculated by using equation (4).

$$RPC_t = \frac{\sum_{t-t_\Delta-1<t'<t} \theta^{t-t'} \cdot RPE_{t'}}{t_\Delta - 1} - \frac{\sum_{t-t_\Delta-1<t'<t} \theta^{t-t'} \cdot RPE_{t'} - RPE_{t'-1}}{t_\Delta - 2} \tag{16}$$

Where $RPE_{t'}$ is the prediction error at time t', t is the time to be predicted, and t_Δ is the size of the observation time window. a is the error affecting the attenuation factor, which reflects that the influence of the prediction error on the final prediction result is attenuated as the distance from the current time increases. The two parts of Eq.(17) correspond to the two components of the correction value $RPC_{t'}$ of the request error: 1) error-weighted average of the overall level of error within the time window and 2) a weighted average of the error change values of the error trend in the time window. Based on the above error correction, the final request index $R_{c,x}$ of service x can be expressed as

$$R_{c,x} = RPE_{t,x} - RPC_{t,x} \tag{17}$$

By ranking the correction value of request index $R_{c,x}$ of the service x recorded in the request log, the fog node can reasonably optimize the cache space.

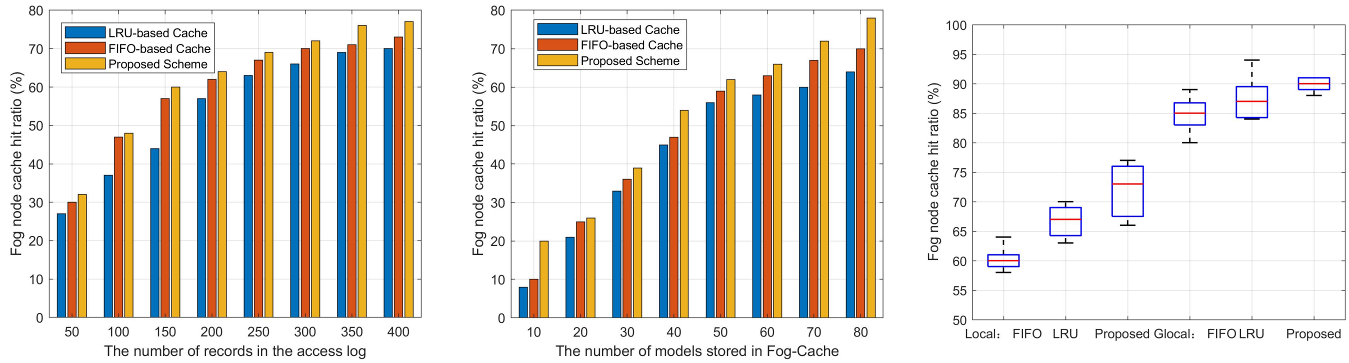
D. Proposed Cache Optimization Algorithm

The principle of cache optimization is to provide a more prioritized cache for AI service requests that are more likely to arrive. Therefore, our proposed algorithm aims to predict the possibility of each coming service based on historical data, and use this as a popularity for cache optimization. We assume that the number of requests is within a reasonable range, and not in special cases (e.g. when Fog nodes encounter a DDoS attack). In this algorithm, we believe that each Fog node has a limited cache space for caching AI models and related data. According to the request index prediction based on PBP and error correction, we can get the above optimization algorithm. The cache space of each node is limited. When the cache space is not fully occupied, the node caches all the received pre-trained models and decentralized big data. When the cache resource is tight, the node predicts and calculates each received request according to the historical data, and ranks the cache priority according to the request index. In addition, since the storage of the request history data also needs to occupy a certain storage resource, We also set a time window for the access log.

V. Simulation and Discussion

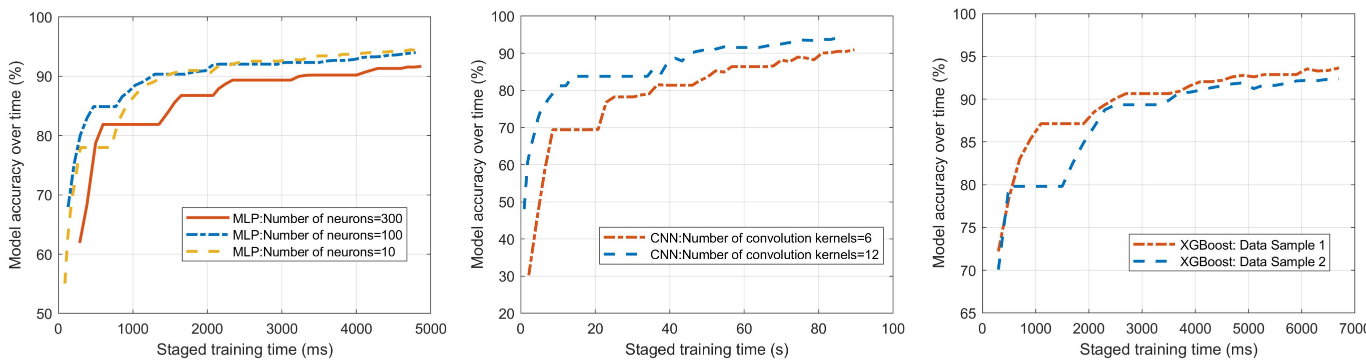
The efficiencies and reasonable explanations of the proposed scheme is given in this section by a series of comparisons including: local cache hit ratio, global cache hit ratio, typical machine learning algorithm convergence time comparison, data set impact on model training, etc.

The proposed Stage Training scheme and cache optimization algorithms are simulated respectively. We establish SD-ICN network and nodes, simulate the behaviors and functions of SD-ICN and AI model training, service sharing and distributing. We simulated an SD-ICN network consisting of multiple fog nodes. Each fog node can cache decentralized big data and pre-trained models and train the matched data and models. The demands of edge users for the pre-trained models are then randomly generated at the edge of the SD-ICN, and the model receive requests from the users. We set



(a) Local and global cache hit ratio (b) Impact of access log data on cache hit ratio (c) The effect of cache size on cache hit ratio

Fig. 4: Comparison of cache hit ratio under different cache optimization methods



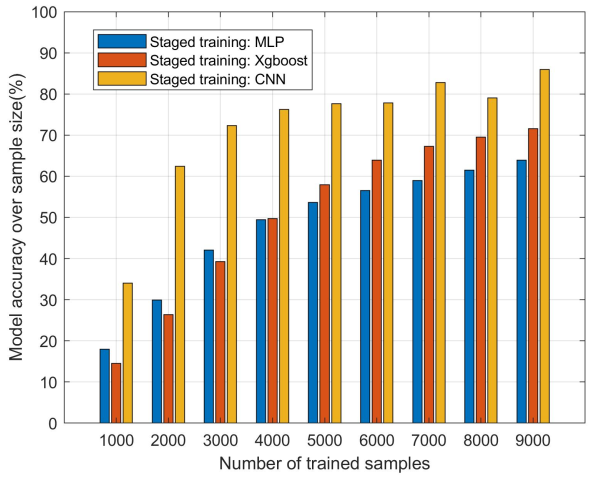
(a) Model accuracy of MLP over time (b) Model accuracy of CNN over time (c) Model accuracy of XGBoost over time

Fig. 5: Comparison of model accuracy for AI service sharing

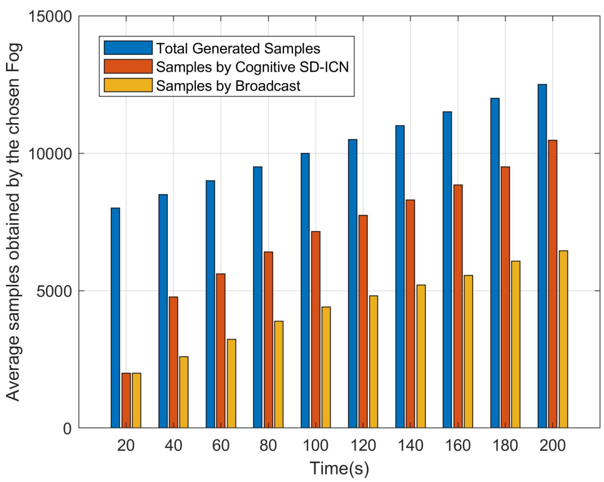
up an access log for each fog node and set a time window to record historical data for requests to the models. In order to better match the actual situation of the network, the demand for a certain model is set to the process of rising first and then falling.

We first simulated the SD-ICN cache optimization. In this set of experiments, our proposed algorithm is compared by the *1) Least Recently Used (LRU)* and *2) First In First Out (FIFO)*. As shown in Fig.**??**(a), We simulated the impact of the size of the recorded access log on popularity prediction. It is obvious that the cache hit rate of the fog node increases as the number of access log records increases. This is because all three algorithms have certain requirements for historical data. However, due to the efficient use of historical data, our proposed algorithm performs best in the three algorithms. In addition, we can also find that the cache hit rate grows slower as the size of the access log increases. This shows that we don't actually need to take up too much storage resources to access the logs. Fig.**??**(b) shows the change in cache hit ratio as the size of the cache space grows. The increase in cache space will undoubtedly lead to an increase in cache hit ratio. The cache hit ratio of our algorithm is higher than LRU and FIFO. Fig.**??**(c) compares the cache hit ratios of the three algorithms locally and globally. If a pre-training model exists in the local cache, then we call it locally available. If the resource is not in the local cache, but we are able to obtain the model through the SD-ICN, the model is globally available. As shown in Fig.**??**(c), the proposed cache optimization algorithm based on popularity prediction is significantly higher in local cache hit ratio than LRU and FIFO. A higher local cache hit rate helps ease network throughput pressure and improve Quality of Experience (QoE) of users. However, in the global cache hit ratio, the FIFO and LRU are only slightly lower than the algorithm proposed in this paper. This shows that the SD-ICN-based AI service sharing mechanism can achieve efficient and fast service caching and distribution.

For the specific algorithms under the proposed scheme, we simulated the case of staged training of a series of typical machine learning algorithms implemented by SD-ICN, including Multi-Layer Perception (MLP), Convolutional Neural Networks (CNN) and eXtreme Gradient Boosting (XGBoost). We validated our scheme using a dataset from Kaggle. We assume that different data samples are continuously generated by users at the edge of the network. When a node requesting samples receives new data samples, the existing AI model will be further trained, thereby improving the accuracy of the model. Fig.**??**(a), shows the accuracy of the MLP model as it changes over time in the SD-ICN. When the model is initially



(a) Model accuracy as the sample size increases



(b) Average obtained samples comparison over time

Fig. 6: The effectiveness of the proposed Cognitive SD-ICN scheme based AI service sharing

requested, since it does not exist in the network, the fog node requests data to train the MLP model. The accuracy of the model rises at an early stage, but most services have high requirements for model accuracy. Therefore, as the model ages in the SD-ICN, the accuracy of the model increases with the staged training. We also found the same rule in CNN and XGBoost, and the accuracy of the model increases with time. The specific accuracy-time diagrams are shown in Fig.**??**(b) and Fig.**??**(c), respectively.

In addition to the above simulations, we also investigated whether the proposed SD-ICN mechanism can be improved over traditional methods. Firstly, we simulated the impact of training data on the algorithm. As decentralized big data continues to emerge, models cached in SD-ICN nodes will also be refined by decentralized big data. As shown in Fig.**??**a, we can see the accuracy improvement of the three AI service models with the increase in the size of the decentralized big data samples. This is consistent with the trend shown in Fig.**??**. Fewer model training samples can easily cause problems such as over-fitting and affect the scope of application of the model. Secondly, we further simulated whether the proposed scheme could obtain the required data samples faster. As shown in Fig.**??**b, we generate data samples in the same network nodes under the same network topology, and compare the speed of Cognitive SD-ICN method with traditional broadcast in obtaining data samples. Under the same initial conditions, the Cognitive SD-ICN method can obviously obtain data samples faster. From the perspective of the final results, the proposed Cognitive SD-ICN method is 62.11% faster than the traditional method.

VI. Conclusion

To realize the intelligent service sharing and task processing, this paper focused on the distribution and caching of efficient AI services in SD-ICN. An AI service distribution architecture for IoT based on integrated decentralized big data and SD-ICN is proposed to share the decentralized big data and pre-trained AI models. In order to provide user request-oriented service popularity model for caching and distribution optimization, we implemented PBP-based popularity prediction by requesting historical data of the service demand of to optimize the cache policy and improve QoS. By the transition from decentralized big data to SD-ICN, the efficient training, distribution, caching and sharing of various AI services are realized at the edge of the network. Simulation results indicate that the proposed cognitive SD-ICN scheme has 62.11% improved to the conventional methods.

References

[1] Q. Zhang, X. Wang, M. Huang, K. Li and S. K. Das, “Software Defined Networking Meets Information Centric Networking: A Survey,” *IEEE Access*, vol. 6, pp. 39547-39563, 2018.
[2] J. Wu, M. Dong, K. Ota, J. Li and Z. Guan, “FCSS: Fog-Computing-based Content-Aware Filtering for Security Services in Information-Centric Social Networks,” *IEEE Transactions on Emerging Topics in Computing*, vol. 7, no. 4, pp. 553-564, 2019.
[3] J. Wu, M. Dong, K. Ota, J. Li, W. Yang and M. Wang, “Fog-Computing-Enabled Cognitive Network Function Virtualization for an Information-Centric Future Internet,” *IEEE Communications Magazine*, vol. 57, no. 7, pp. 48-54, 2019.
[4] T. Mick, R. Tourani, and S. Misra, “LASeR: Lightweight Authentication and Secured Routing for NDN IoT in Smart Cities,” *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 755-764, 2018.
[5] S. Arshad, M. A. Azam, M. H. Rehmani, and J. Loo, “Recent Advances in Information-Centric Networking-Based Internet of Things (ICN-IoT),” *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 2128-2158, 2019.
[6] M. Mohammadi, A. Al-Fuqaha, S. Sorour, and M. Guizani, “Deep Learning for IoT Big Data and Streaming Analytics: A Survey,” *IEEE Communications Surveys & Tutorials*, vol. 20, no. 4, pp. 2923-2960, 2018.
[7] A. Yousefpour, G. Ishigaki, R. Gour, and J. P. Jue, “On Reducing IoT Service Delay via Fog Offloading,” *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 998-1010, 2018.
[8] J. Wu, M. Dong, K. Ota, J. Li, W. Yang, ”Application-Aware Consensus Management for Software-defined Intelligent Blockchain in IoT,” IEEE Network, vol. 34, no. 1, pp. 69-75, 2020.
[9] G. Han, X. Yang, L. Liu, S. Chan, and W. Zhang, “A Coverage-Aware Hierarchical Charging Algorithm in Wireless Rechargeable Sensor Networks,” *IEEE Network Magazine*, DOI: 10.1109/MNET.2018.1800197, 2018.
[10] Y. Zhang , R. Gravina, H. Lu M. Villari, and G. Fortino, “PEA: Parallel Electrocardiogram-based Authentication for Smart Healthcare Systems,” *Journal of Network and Computer Applications*, vol. 117, no. 4, pp. 10-16, 2018.

[11] H. Lu, Y. Li, S. Mu, D. Wang, H. Kim and S. Serikawa, "Motor Anomaly Detection for Unmanned Aerial Vehicles Using Reinforcement Learning," *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 2315-2322, Aug. 2018.
[12] G. Han, H. Guan, J. Wu, S. Chan, L. Shu, and W. Zhang, "An Uneven Cluster-Based Mobile Charging Algorithm for Wireless Rechargeable Sensor Networks," *IEEE Systems Journal*, DOI: 10.1109/JSYST.2018.2879084, 2018.
[13] X. Chen, S. Tang, Z. Lu, J. Wu, Y. Duan, S, Huang, and Q. Tang, "iDiSC: A New Approach to IoT-Data-Intensive Service Components Deployment in Edge-Cloud-Hybrid System," *IEEE Access*, vol. 7, pp. 59172-59184, 2019.
[14] D. C. Yacchirema, D. Sarabia-Jcome, C. E. Palau, and M. Esteve, "A Smart System for Sleep Monitoring by Integrating IoT With Big Data Analytics," *IEEE Access*, DOI: 10.1109/ACCESS.2018.2849822, 2018.
[15] K. Wang, C. Xu, Y. Zhang, S. Guo, and A. Y. Zomaya, "Robust Big Data Analytics for Electricity Price Forecasting in the Smart Grid," *IEEE Transactions on Big Data*, vol. 5, no. 1, pp. 34-45, 2019.
[16] M. A. Alsheikh, D. Niyato, S. Lin, H. Tan, and Z. Han, "Mobile Big Data Analytics Using Deep Learning and Apache Spark," *IEEE Network*, vol. 30, no. 3, pp. 22-29, 2016.
[17] X. Lin, J. Li, J. Wu, H. Liang, W. Yang, Making Knowledge Tradable in Edge-AI Enabled IoT: A Consortium Blockchain-based Efficient and Incentive Approach, IEEE Transactions on Industrial Informatics, vol. 15, no. 12, pp. 6367-6378, 2019.
[18] W. Liu, J. Zhang, Z. Liang, L. Peng, and Jun Cai, "Content Popularity Prediction and Caching for ICN: A Deep Learning Approach With SDN," *IEEE Access*, vol. 6, no. 1, pp. 5075-5089, 2017.
[19] J. Lv, X. Wang, and Min Huang, "Ant Colony Optimization-Inspired ICN Routing with Content Concentration and Similarity Relation," *IEEE Communications Letters* vol. 21, no. 6, pp. 1313-1316, 2017.
[20] H. Lu et al., "CONet: A Cognitive Ocean Network," *IEEE Wireless Communications*, vol. 26, no. 3, pp. 90-96, June 2019.
[21] H. Khedher, H. Afifi, and H. Moustafa, "Optimal placement algorithm (OPA) for IoT over ICN," *2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, DOI: 10.1109/INFCOMW.2017.8116405, 2017.
[22] Z. Zhang, C. Lung, I. Lambadaris, and M. St-Hilaire, "IoT Data Lifetime-Based Cooperative Caching Scheme for ICN-IoT Networks," *IEEE International Conference on Communications (ICC)*, DOI: 10.1109/ICC.2018.8422100, 2018.
[23] Q. Wang, B. Lee, N. Murray, and Y, Qiao, "MR-IoT: An information centric MapReduce framework for IoT," *15th IEEE Annual Consumer Communications & Networking Conference (CCNC)*, DOI: 10.1109/CCNC.2018.8319184, 2018.
[24] S. Arshad, M. A. Azam, M. H. Rehmani and J. Loo, "Recent Advances in Information-Centric Networking-Based Internet of Things (ICN-IoT)," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 2128-2158, 2019.
[25] Z. Tan, Y. Wang, Y. Zhang, and J. Zhou, "Novel Time Series Approach for Predicting the Long-Term Popularity of Online Videos," *IEEE Transactions on Broadcasting*, vol. 62, no. 2, pp. 436-445, 2016.
[26] S. Sheu, Y. Chen, C. Chang, and Z. G. Zhang, "A Block Replacement Policy for Systems Subject to Non-homogeneous Pure Birth Shocks," *IEEE Transactions on Reliability*, vol. 61, no. 3, pp. 741-748, 2012.

Siyi Liao received the B.S. degree from School of Electronic Information Engineering in Beijing Jiao Tong University, Beijing, China, in 2017. Now, he is a candidate Ph.D. at School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai, China. His research interests are focusing on Multi-access Edge Computing (MEC), Internet of Vehicles (IoV), etc. He is a student member of IEEE.

Jun Wu (S'08-M'12) received the Ph.D. degree in information and telecommunication studies from Waseda University, Japan, in 2011. He was a Post-Doctoral Researcher with the Research Institute for Secure Systems, National Institute of Advanced Industrial Science and Technology (AIST), Japan, from 2011 to 2012. He was a Researcher with the Global Information and Telecommunication Institute, Waseda University, Japan, from 2011 to 2013. He is currently an associate professor of School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, China. He is also the vice director of National Engineering Laboratory for Information Content Analysis Technology, Shanghai Jiao Tong University, China. He is the chair of IEEE P21451-1-5 Standard Working Group. He has hosted and participated in a lot of research projects including National Natural Science Foundation of China (NFSC), National 863 Plan and 973 Plan of China, Japan Society of the Promotion of Science Projects (JSPS), etc. His research interests include the advanced computing, communications and security techniques of software-defined networks (SDN), information-centric networks (ICN) smart grids, Internet of Things (IoT), 5G, etc., where he has published more than 120 refereed papers. He has been the Track Chair of VTC 2019 and the TPC Member of more than ten international conferences including ICC, GLOBECOM, WINCON, etc. He has been a Guest Editor of the IEEE Sensors Journal, Sensors, ICT Express. He is an Associate Editor of the IEEE Access.

Jianhua Li is a professor/Ph.D. supervisor and the dean of School of Information Security Engineering, Shanghai Jiao Tong University, Shanghai, China. He got his BS, MS and Ph.D. degrees from Shanghai Jiao Tong University, in 1986, 1991 and 1998, respectively. He was the chief expert in the information security committee experts of National High Technology Research and Development Program of China (863 Program) of China. He is the member of the committee of information security area of the state 10th five-year plan of China. Also, he is a committee expert of China State Secrecy Bureau and Shanghai Secrecy Bureau. He was the leader of more than 30 state/province projects of China, and published more than 200 papers. He published 6 books and has about 20 patents. He made 3 standards and has 5 software copyrights. He got the Second Prize of National Technology Progress Award of China in 2005. He got the First Prize of National Technology Progress Award of Shanghai in 2003 and 2004, and he got two First Prize of National Technology Progress Awards of Shanghai in 2004. His research interests include cyberspace security, Next Generation Networks, etc.

Ali Kashif Bashir is a Senior Lecturer at the Department of Computing and Mathematics, Manchester Metropolitan University, United Kingdom. He is also with School of Electrical Engineering and Computer Science, National University of Science and Technology, Islamabad (NUST), Islamabad, Pakistan. He is a senior member of IEEE and Distinguished Speaker of ACM. His past assignments include Associate Professor of Information and Communication Technologies, Faculty of Science and Technology, University of the Faroe Islands, Denmark; Osaka University, Japan (71 in QS Ranking 2020); Nara National College of Technology, Japan; the National Fusion Research Institute, South Korea; Southern Power Company Ltd., South Korea, and the Seoul Metropolitan Government, South Korea. He is also advising several startups in the field of STEM based education, robotics, and smart homes. Currently, he is supervising RNS Solutions, a multinational organization as Chief Research Officer. He received his Ph.D. in computer science and engineering from Korea University (83 in QS Ranking 2020) South Korea. MS from Ajou University, South Korea and BS from UMT, Pakistan. He is author of over 80 peer-reviewed articles. He is supervising/co-supervising several graduate (MS and PhD) students. His research interests include internet of things, wireless networks, distributed systems, network/cyber security, cloud/network function virtualization, etc. He is serving as the Editor-in-chief of the IEEE FUTURE DIRECTIONS NEWSLETTER.

Alireza Jolfaei is an Assistant Professor of Cyber Security in the Department of Computing at Macquarie University. Prior to this appointment, He worked as an Assistant Professor at Federation University Australia and Temple University in Philadelphia, USA. His research is centred around Cyber Security, IoT Security, Human-in-the-Loop CPS Security, Cryptography, AI and Machine Learning for Cyber Security. He received the bachelors (Hons.) degree in biomedical engineering from Islamic Azad University, Tehran, Iran, in 2008, and the masters (Hons.) degree in telecommunication engineering from Imam Hossein Comprehensive University, and completed his PhD in Applied Cryptography at Griffith University. Throughout his academic journey, he have been awarded multiple times for Academic Excellence, University Contribution, and Inclusion and Diversity Support. He received the prestigious IEEE Australian Council Award for his paper published at IEEE Transactions on Information Forensics and Security. He served as the guest associate editor of IEEE journals and transactions, including IEEE IoT Journal, IEEE Transactions on Industrial Applications, and IEEE Transactions on Intelligent Transportation Systems. He served over 10 conferences in leadership capacities including program co-Chair, track Chair, session Chair, and Technical Program Committee member, including IEEE TrustCom and DependSys. He chaired the Computational Intelligence Society in the IEEE Victorian Section and the professional and career activities for the IEEE Queensland Section. He is a Senior Member of IEEE and a distinguished speaker of ACM on the topic of cyber-physical systems security.

Shahid Mumtaz has more than 10 years of wireless industry/academic experience and is currently working as Principal Research Scientist and Technical Manager at Instituto de Telecomunicaes (IT) Portugal. He has received his Master and PhD degrees in Electrical & Electronic Engineering from Blekinge Institute of Technology, Sweden, and University of Aveiro, Portugal in 2006 and 2011, respectively. From 2005 to 2006 for Ericsson and Huawei at Research Labs in Sweden as Researcher. Since January 2015, he has been with the Department of Electrical and Electronic Engineering at the University of Aveiro, Portugal where he is Adjunct Associate Professor and Director of the Wireless Networking and Communications Group. He uses mathematical and system level tools to model and analyze emerging wireless communication architectures, leading to innovative and/or theoretically optimal new communication techniques. He is working closely with leading R&D groups in the industry to transition these ideas to practice. His work is currently supported by Samsung, Huawei, and Ericsson, as well as FCT. He has also been providing consultancy to Intel, Huawei, and Portugal Telecom. In January 2017, IEEE has started a new standard on P1932.1: Standard for Licensed/Unlicensed Spectrum Interoperability in Wireless Mobile Networks and was elected as Vice chair for this standard. Furthermore, in January 2015, he was also elected Vice-Chair of IEEE Research Project on Vision of Green Standardization due to his pioneer work in green communication as well. He is also actively involved in 3GPP standardization on LTE release 12 onwards, along with major manufacturers (Huawei and Intel).Dr Mumtaz has published 3 books and more than 150 publications in very high-rank IEEE transactions, journals, book chapters, international conferences. He is serving as Scientific Expert and Evaluator for Research Funding Agencies, such as European Commission, and COST.

Nida Kvedarait is an associate professor in Kaunas University of Technology who holds a PhD degree in Educational Sciences. She published more than 50 research papers and is the co-author of 3 books. The researcher is the member of editorial board of scientific periodicals and scientific committees of international conferences; an expert of evaluating research projects proposals; an active participant of the scientific conferences both on international and national levels..