A Comprehensive Survey of Recent Proposed Content Replacement Strategies for Cooperative Edge Caching in IoT

Fariba Majidi[1, 2]
1- Department of Computer Engineering, Mobarakeh Branch, Islamic Azad University, Isfahan, Iran.
Email: F.majidi@khuisf.ac.ir (Corresponding author)
2- Department of Computer Engineering, Isfahan (Khorasgan) Branch, Islamic Azad University, Isfahan, Iran.

ABSTRACT:
The Internet of Things significantly increases the number of terminals and network traffic load, while its real-time applications require minimal latency to access the requested contents from data centers. Despite the processing and storage capabilities of base stations in 5G networks, the use of edge caching has proven to be an effective solution to reduce content access delay and repetitive traffic. This optimization of content transfer through the internet is crucial for maintaining the efficiency and performance of IoT applications. However, several challenges must be addressed. The limited storage resources, the constant changes in network nodes, and the dynamic patterns of content requests and user behavior pose fundamental challenges to content placement strategies. Developing an effective content placement strategy requires a comprehensive understanding of these factors and the ability to adapt to the evolving network environment.. In this paper, Challenges and issues facing content placement strategies in cooperate edge caching are described and recent researches in this field are reviewed, categorized, and explained comprehensively.

KEYWORDS: Internet of Things, Edge caching, Edge computing, Deep Reinforcement Learning, Federate learning

1. INTRODUCTION

In recent years, With the advent of the Internet of Things (IoT), phones are not the only mobile terminals transmitting data, but numerous varieties of sensors are connected too. Therefore, information is not exchanged only between humans, but also is extended to communication between objects and can fulfill needs such as identification, location, tracking, management, and intelligent monitoring. Therefore, IoT significantly increases the number of terminals and network traffic load. IoT systems have presented new needs for cloud computing-based solutions. Especially in some real-time applications, such as smart cars, real-time processing of sensed data and executing the reactions is indispensable. Although the delay of communications within the local network is low, accessing the content available in data centers through the Internet has posed an important challenge to these systems [2].

Edge computing has brought data storage, computing, and control closer, in the edge devices rather than in a central cloud server. Therefore, each edge device plays its role in determining what information should be stored or processed locally and what should be retrieved from the cloud server. Edge computing has perfected the IoT in high scalability, low latency, location awareness, and real-time use of local devices' computing capabilities. As presented in [33], most network traffic for IoT systems consists of frequent requests for duplicate content. The continuous transfer of these popular contents will cause a significant increase in network traffic redundancy [65]. Therefore, by storing them in the cache of edge devices, the contents would be provided to end users without repeated transmission from the central servers [3]. Fig.1 shows how Edge caching would be implemented in a hybrid network. However, edge caching needs

Paper type: Research paper
https:// 10.71822/mjtd.2024.1126829
Received: 22 July 2024; revised: 5 August 2024; accepted: 11 September 2024; published: 1 December 2024
How to cite this paper: F. Majidi, **"A Comprehensive Survey of Recent Proposed Content Replacement Strategies for Cooperative Edge Caching in IoT"**, *Majlesi Journal of Telecommunication Devices*, Vol. 13, No. 4, pp. 191-197, 2024.

some considerations. One of the issues is managing the resource utilization. The memory and processing resources of edge devices are limited. So, each node can only store a small amount of media content. Another important issue is network reliability which determines whether a network remains functional when its elements fail at random [5].

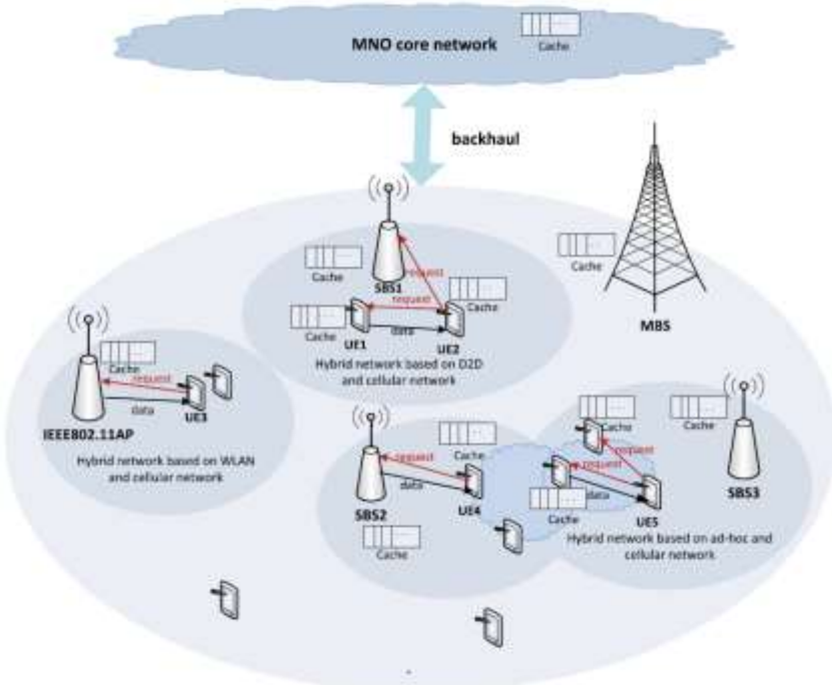

Fig. 1. Edge caching in a hybrid network [4].

Wide and fast communication of edge devices in the 5G network, especially interconnected BSs using the wired mesh network, has made caching possible cooperatively and intelligently. By cooperating between edge devices, it is possible to respond to content requests with local communications between BSs and users, significantly reducing content access delay, also it optimizes resource utilization and reliability.

In all approaches for increasing the performance of edge caching, content replacement strategy has been the focus of the recent edge cache research. If the popular contents cached in the edge devices are chosen accurately for feature requests, it will improve the hit rate of the edge cache. Due to the limitation of the cache space, a solution must be provided to select the contents for eviction when the cache is full. So that there would be enough space to insert new popular content. The replacement strategy is usually based on content popularity or the cost of storing and transferring them. In this paper, we survey recent state-of-the-art literature on cooperate edge caching and taxonomy based on the methods utilized for content replacement strategy as optimization methods and intelligent methods.

For the rest of this paper, we first describe the research that used optimization methods for solving the problem of content replacement and categorize them by the metrics they used for optimization formulas. Then the recently proposed intelligent methods as content replacement strategies are explained. The research that utilizes the most trends machine learning approach is classified in this section. Finally, we present the future outlook for this area and conclude the article.

2. OPTIMIZATION METHODS

In the optimization methods, to determine the appropriate content replacement strategy in the edge cache, the factors and problems to be optimized must be carefully selected. Also, the method of problem formulation has a great impact on the practical application and computational complexity of the proposed method because neglecting an important limitation can lead to solutions that are not applicable in practice [35]. The performance of edge cache is usually evaluated from two aspects: QoE and network performance. In most cases, hit rate and content access delay metrics are used for the QoE evaluations, also local network traffic load and request load on the central server, are considered to evaluate network efficiency. In the collaborative method, the classification factors Influence the proposed strategy considerably. In the rest of this section, content access delay, cache hit rate, resource utilization, cost, and classification factors are described as metrics mostly used in optimization methods.

2.1. Content access delay

One of the most important metrics to evaluate the efficiency of edge caching is the Content access delay [35]. Although this parameter depends on the processing time, but the network delay exerts a substantial impact on it. In the article [12], Saputra et al try to reduce the delay by formulating the contents that are stored in the cache and deciding how the devices access the content, then this problem has been converted into MINLP and it is solved with the divide and conquer algorithm. In [15], the problem is formulated as MILP, and in [20], an algorithm using belief propagation is presented to solve the proposed problem. In [37], the problem has been formulated as a multi-agent multi-armed bandit and has used replacement to solve this problem. In the article [38], the optimization problem of content storage in edge cache is formulated. So, the delay of content requests of all users in MEC networks could be minimized.

2.2. Edge cache hit rate

The cache hit rate is the probability that the content requested by the users exists in the cache of the edge devices [39]. In other words, the percentage of requests that will be fulfilled by the cache. Improving this parameter often has a positive effect on other parameters as well. Hence, optimization of cache hit rate has been considered as the main issue in many researches. In [14], the problem is formulated as MDP, and the solution is proposed by the placement strategy. In [17], Lagrangian multiplication and zero and one knapsack methods have been used to solve the problem, and in [7], the optimization of the hit rate of the edge cache in the Vehicular content network has been investigated, and in [40], the hit rate is optimized in three layers (routers, base stations, and users) also as interlayer and extra layer.

2.3. Cost

By reducing energy consumption, increasing power, and improving QoE, the costs of service providers are reduced and they can earn more with the same workload [39]. Answering popular requests locally also allows higher layers to handle more requests, improving overall performance. So, [44] solved the problem using graph coloring. In [41], one of the important optimization factors is energy, which is formulated as ILP. In [45], the optimization of the transmission cost is achieved using the PSO algorithm.

2.4. Device classification factors

As it was mentioned before, the classification BSs has a great impact on the efficiency of the collaborative edge cache, therefore [1], optimize the caching performance by finding the optimal distance between the base stations of a group and reducing the cooperation costs between them, and [11] has optimized the size of groups of cooperating BSs in their proposed merging optimization method.

It should be noted that due to the high complexity of edge deployment and the fully dynamic conditions of wireless networks, it is difficult to find a design-based solution [22]. However, even though some of the proposed methods that used optimization to improve QoE, have achieved good results, these methods in most cases have overlooked the long-term impact of current decisions. Therefore, a considerable number of the presented algorithms operate optimally or close to optimally in the system only for a specific period of time. Therefore, these methods cannot fully and effectively improve the efficiency of the edge cache [9]. A secondary section heading is enumerated by a capital letter followed by a period and is flush left above the section. The first letter of each important word is capitalized and the heading is italicized.

3. INTELLIGENT METHODS

Designing a suitable strategy for replacing contents in the collaborative edge cache requires considering the features and complexities of edge networks and the continuous mobility of wireless devices. If the proposed method relies on specific information as input, it may be difficult to obtain this information due to the great diversity in border channels and devices as well as security policies.

Another issue that should be considered when designing an algorithm for content replacement is the dynamics of content popularity. In each set of cooperating edge devices, the popularity of each content changes continuously, and content may be popular only in a short period of time [1]. Therefore, the placement strategy is optimal when it is based on the correct distribution of content popularity, by which it can recognize the content that will be requested in the near future. But predicting content popularity is difficult for several reasons:

The contents are requested by different users, so the popularity of the contents changes with the mobility of the local network users covered by each base station. Another problem is that users' interests vary in different situations (location, network topology, personality traits, etc.).

Recently, edge devices have higher computing and storage capabilities [9]. Therefore, it is possible to implement some methods such as massive data analysis and deep learning on these devices, and it is also possible to apply artificial intelligence techniques in mobile edge networks to understand user behaviors and network characteristics. With an understanding of the user and the network, patterns can be designed that are context-aware and intelligent, allowing edge devices to make the right decision at the right time to choose what content to store in their limited cache resources [9].

Many recent researches have used strategies based on various artificial intelligence and machine learning methods for edge cache management. [4], [17], [24], [26]– [28], [46] In the next sections, methods using reinforcement learning, deep reinforcement learning, and federated learning for predicting user requests are expressed.

3.1. Reinforcement learning

Reinforcement learning is concerned with how an agent in an environment should act to maximize overall reward. This method is not a subset of any of the supervised and unsupervised methods [40]. The environment is usually modeled using Markov Decision Process (MDP), but instead of a precise mathematical model of MDP, the target of MDP is usually very large, and precise mathematical methods cannot be used for its design. فلانی وهمکاران [47], using a history of content requests and multi-agent reinforcement learning, presented a method for replacing content in a collaborative edge cache.

3.1. Deep reinforcement learning

The use of reinforcement learning has allowed agents to solve the decision problem by learning from interacting with the environment. To achieve this goal, the environment must be determined in a suitable way and with an acceptable complexity, which makes the use of this method limited only to cases where the characteristics of the environment can be extracted. With the advent of deep neural networks, agents can learn some compact representations using high-dimensional and raw data. Deep learning is a type of artificial neural network that mimics the way the human brain works in data processing and pattern creation. Some recent research has used deep neural networks to model requested content. [6], [20], [28], [48] By combining deep learning and reinforcement learning, a powerful model can be created that has the ability to solve the great parts of previously intractable problems. Therefore, agents can have optimal control over the environment using the knowledge they obtain directly from the raw data. Researchers in [6], [13], [18], [19], [34], [28]–[31], [49], [50] utilize distributed and multi-agent deep reinforcement learning and have been able to significantly improve the hit rate by improved predicting the content in the collaborative edge cache.

3.2. Federated deep reinforcement learning

Another technique that has been proposed with the aim of improving security and reducing data transfer overhead in edge cache is federated learning. Federated learning is a method that uses agents that have local data samples to perform learning in a decentralized manner without transferring data to a central server or other agents. DRL techniques require a high computational capacity of resources to find optimal solutions. In particular, if there are a large number of resource optimization factors, parameters, and criteria for resource optimization in large-scale MEC systems (operator networks in cities), advanced distributed deep learning (DL) methods should be used for an operable method in the real world.

As shown in Fig.2, although maintaining and training each edge device as a DRL agent could increase the performance, the use of distributed DRL is only practical in MEC systems because of insufficient time and data for large-scale training. Also, most distributed DRL architectures cannot handle unbalanced and Non-IID data, and they can't support privacy issues too [32].

Therefore, the researchers in [8], [9], [22], [51] have used federated learning to train DRL agents. Because federated learning can overcome the following challenges as described in the following [32].

- Non-uniform distribution of data (Non-IID): The training data in the edge devices is based on the environment it has experienced. Each edge device has its own processing capability and energy consumption. Therefore, the local data of each of them cannot be a suitable representative for the training data of all end users and edge devices. In federated learning, this challenge can be overcome by integrating updated models with FedAvg. [32]
- Limited communications: Users often go offline unpredictably or have poor communication resources. Federated learning requires only a fraction of users to upload their updates during a training session, so it can handle situations where clients often go offline unpredictably [9].
- Imbalance: Some edge devices may have more computational tasks and some may experience more states of mobile networks, resulting in different amounts of training data among them. This challenge can be dealt with by the FedAvg algorithm [9].
- Privacy and Security: The amount of information that needs to be loaded for federated learning is the minimum volume of updates that can be used to improve the behavior of the DRL agent. In addition, different privacy-preserving and secure aggregation techniques can be applied, which prevent the inclusion of privacy-sensitive information in local updates [32].

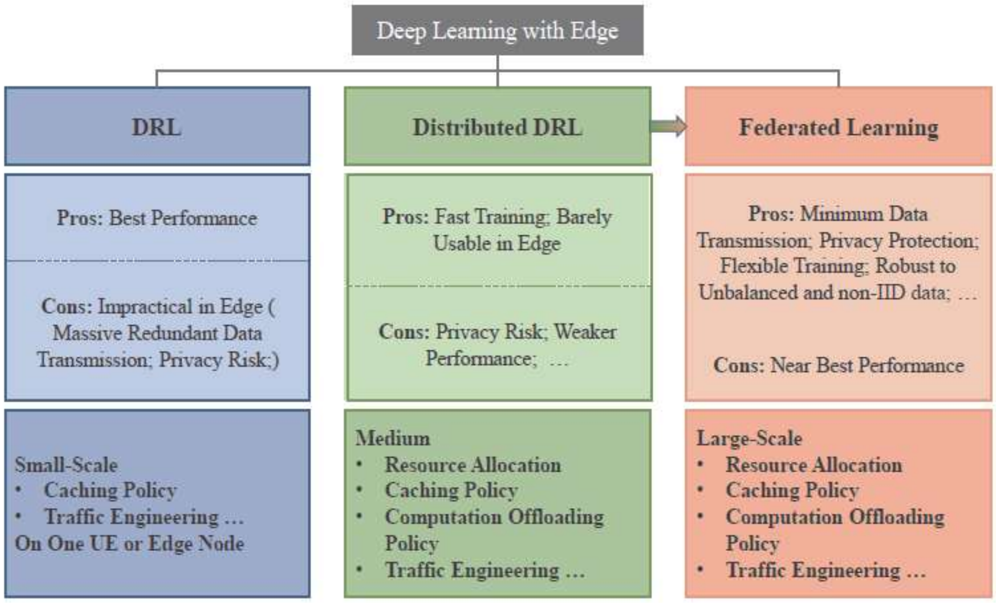

Fig. 2. Comparison of centralized, distributed, and federated methods in using DRL for border cache management [9].

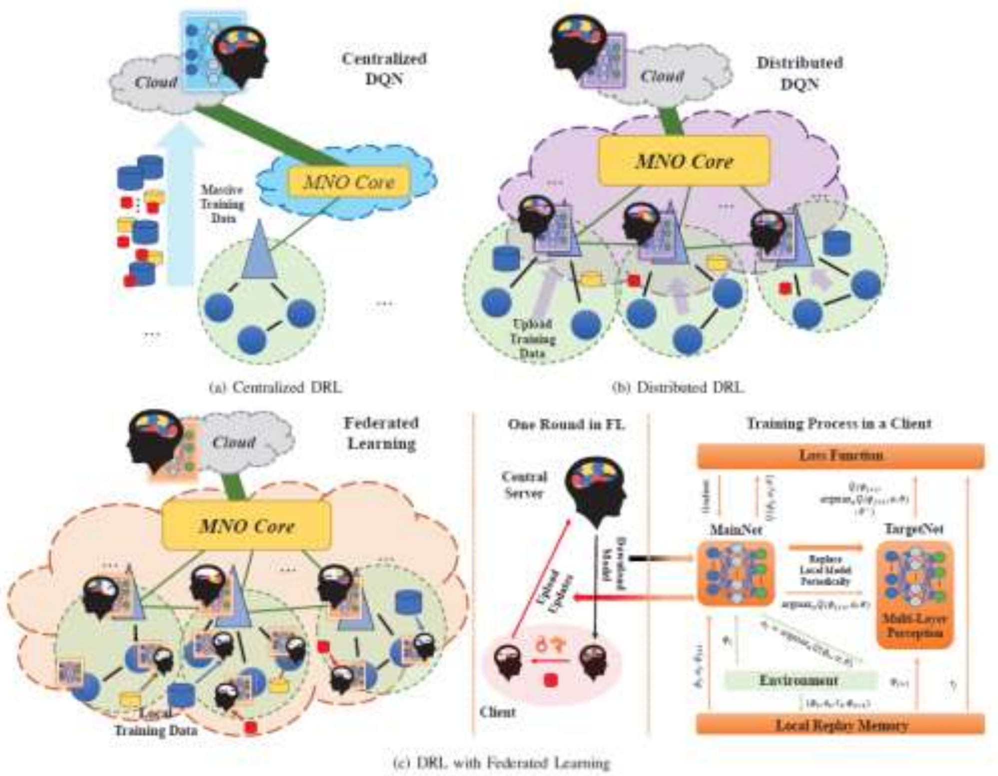

Fig. 3. Modeling of centralized, distributed and federated DRL methods presented in [9].

[8]-[9], [22] have proposed a method by combines federated learning and deep reinforcement learning. In these methods, each base station implements deep reinforcement learning using its local data and sends only the updated model to the central server. By aggregating the models received from a set of agents, the central server creates an updated model and returns it to all active agents to continue learning and predict popular content that they should store in the

collaborative edge cache. Figure 3-3 compares the method presented in [9] with centralized and distributed DRL methods. Fig.3 compares the method presented in [9] with centralized and distributed DRL methods.

4. CONCLUSIONS

Recent research reviewed in this paper has shown two main challenges for improving cooperate edge cache performance. Correctly predicting future user requests with the lowest processing cost is the first and most important challenge. To overcome this challenge, despite the existing capacities of the current edge networks, the best solution is to use intelligent methods to determine the most suitable content replacement strategy in the edge cache. The second important challenge is the effective cooperation of edge devices in storing and responding to user requests. The numerous edge devices could utilize their storage for more variation of contents so respond locally to a relatively large number of user requests. However, their capability to cooperate in caching popular contents would increase the edge cache hit rate and pose a significant challenge to finding an optimal method for Determining cooperating groups and their cooperation methods so that, in addition to improving the hit rate of the cache, it does not impose too much traffic overhead on the local network, which slows down the user's access to content and causes longer delays. The problem of providing privacy and network security is also a challenge that should always be considered in multiple-device cooperation in a public network.

REFERENCES

[1] L. Yang, Y. Chen, L. Li, and H. Jiang, Cooperative Caching and Delivery Algorithm Based on Content Access Patterns at Network Edge, vol. 278. Springer International Publishing, 2019.

[2] Z. Piao, M. Peng, Y. Liu, and M. Daneshmand, “Recent Advances of Edge Cache in Radio Access Networks for Internet of Things: Techniques, Performances, and Challenges,” IEEE Internet Things J., vol. PP, no. c, p. 1, 2018.

[3] X. Wang, M. Chen, T. Taleb, A. Ksentini, and V. C. M. Leung, “Cache in the air: Exploiting content caching and delivery techniques for 5G systems,” IEEE Commun. Mag., vol. 52, no. 2, pp. 131–139, 2014.

[4] G. Xu et al., “An Incremental Learning Based Edge Caching System : From Modeling to Evaluation,” vol. 8, pp. 12499–12509, 2020.

[5] F. Majidi, M. R. Khayyambashi and B. Barekatain, "HFDRL: An Intelligent Dynamic Cooperate Cashing Method Based on Hierarchical Federated Deep Reinforcement Learning in Edge-Enabled IoT," in IEEE Internet of Things Journal, vol. 9, no. 2, pp. 1402-1413, 15 Jan.15, 2022, doi: 10.1109/JIOT.2021.3086623.

[6] R. Wang, M. Li, L. Peng, Y. Hu, M. M. Hassan, and A. Alelaiwi, “Cognitive multi-agent empowering mobile edge computing for resource caching and collaboration,” Futur. Gener. Comput. Syst., vol. 102, pp. 66–74, 2020.

[7] Z. Su, Y. Hui, Q. Xu, T. Yang, J. Liu, and Y. Jia, “An edge caching scheme to distribute content in vehicular networks,” IEEE Trans. Veh. Technol., vol. 67, no. 6, pp. 5346–5356, 2018.

[8] Z. Yu et al., “Federated Learning Based Proactive Content Caching in Edge Computing,” 2018 IEEE Glob. Commun. Conf. GLOBECOM 2018 - Proc., 2018.

[9] X. Wang, Y. Han, C. Wang, Q. Zhao, X. Chen, and M. Chen, “In-Edge AI: Intelligentizing Mobile Edge Computing, Caching and Communication by Federated Learning,” IEEE Netw., vol. XX, pp. 1–10, 2019.

[10] Z. Zheng, L. Song, Z. Han, G. Y. Li, and H. V. Poor, “A stackelberg game approach to proactive caching in large-scale mobile edge networks,” IEEE Trans. Wirel. Commun., vol. 17, no. 8, pp. 5198–5211, 2018.

[11] S. Zhang, P. He, K. Suto, P. Yang, L. Zhao, and X. Shen, “Cooperative Edge Caching in User-Centric Clustered Mobile Networks,” IEEE Trans. Mob. Comput., vol. 17, no. 8, pp. 1791–1805, 2018.

[12] Y. M. Saputra, H. T. Dinh, D. Nguyen, and E. Dutkiewicz, “A Novel Mobile Edge Network Architecture with Joint Caching-Delivering and Horizontal Cooperation,” IEEE Trans. Mob. Comput., pp. 1–1, 2019.

[13] Y. M. Saputra, D. T. Hoang, D. N. Nguyen, E. Dutkiewicz, D. Niyato, and D. I. Kim, “Distributed Deep Learning at the Edge: A Novel Proactive and Cooperative Caching Framework for Mobile Edge Networks,”

[14] C. Wang, S. Wang, D. Li, X. Wang, X. Li, and V. C. M. Leung, “Q-learning based edge caching optimization for D2D enabled hierarchical wireless networks,” Proc. - 15th IEEE Int. Conf. Mob. Ad Hoc Sens. Syst. MASS 2018, pp. 55–63, 2018.

[15] N. Wang, G. Shen, S. K. Bose, and W. Shao, “Zone-Based Cooperative Content Caching and Delivery for Radio Access Network with Mobile Edge Computing,” IEEE Access, vol. 7, pp. 4031–4044, 2019.

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

[17] X. Zhao, P. Yuan, H. Li, and S. Tang, “Collaborative edge caching in context-aware device-to-device networks,” IEEE Trans. Veh. Technol., vol. 67, no. 10, pp. 9583–9596, 2018.

[18] X. He, K. Wang, and W. Xu, “in Edge-Enabled IoT,” IEEE Comput. Intell. Mag., vol. 14, no. NOVEMBER, pp. 10–20, 2019.

[19] G. Qiao, S. Leng, S. Maharjan, Y. Zhang, and S. Member, “Deep Reinforcement Learning for Cooperative Content Caching in Vehicular Edge Computing and Networks,” IEEE Internet Things J., vol. 7, no. 1, pp. 247–257, 2020.

[20] Y. Zhang, Y. Li, R. Wang, J. Lu, X. Ma, and M. Qiu, “PSAC: Proactive Sequence-aware Content Caching via Deep Learning at the Network Edge,” IEEE Trans. Netw. Sci. Eng., vol. 4697, no. c, pp. 1–1, 2020.

[21] W. Chien, H. Weng, and C. Lai, “Q-learning based collaborative cache allocation in mobile edge computing,” Futur. Gener. Comput. Syst., vol. 102, pp. 603–610, 2020.

[22] X. Wang, C. Wang, X. Li, V. C. M. Leung, and T. Taleb, "Federated Deep Reinforcement Learning for Internet of Things with Decentralized Cooperative Edge Caching," IEEE Internet Things J., no. c, pp. 1–1, 2020.

[23] P. Wu, J. Li, L. Shi, M. Ding, K. Cai, and F. Yang, "Dynamic Content Update for Wireless Edge Caching via Deep Reinforcement Learning."

[24] A. Mehrabi, M. Siekkinen, and A. Yla-Jaaski, "QoE-Traffic Optimization Through Collaborative Edge Caching in Adaptive Mobile Video Streaming," IEEE Access, vol. 6, pp. 52261–52276, 2018.

[25] E. Bastug, M. Bennis, and M. Debbah, "Living on the edge: The role of proactive caching in 5G wireless networks," IEEE Commun. Mag., vol. 52, no. 8, pp. 82–89, 2014.

[26] M. F. Pervej, L. T. Tan, and R. Q. Hu, "Artificial Intelligence Assisted Collaborative Edge Caching in Small Cell Networks," arXiv, 2020.

[27] N. Garg et al., "Online Content Popularity Prediction and Learning in Wireless Edge Caching," vol. 68, no. 2, pp. 1087–1100, 2020.

[28] H. Pang, J. Liu, X. Fan, and L. Sun, "Toward Smart and Cooperative Edge Caching for 5G Networks: A Deep Learning Based Approach," 2018 IEEE/ACM 26th Int. Symp. Qual. Serv. IWQoS 2018, 2019.

[29] J. Zheng et al., "Smart Edge Caching-Aided Partial Opportunistic Interference Alignment in HetNets," Mob. Networks Appl., vol. 25, no. 5, pp. 1842–1850, 2020.

[30] M. C. Gursoy, C. Zhong, and S. Velipasalar, "Deep Multi-Agent Reinforcement Learning for Cooperative Edge Caching," pp. 439–457, 2020.

[31] F. Wang, F. Wang, J. Liu, R. Shea, and L. Sun, "Intelligent Video Caching at Network Edge : A Multi-Agent Deep Reinforcement Learning Approach," 2017.

[32] H. Brendan McMahan, E. Moore, D. Ramage, S. Hampson, and B. Agüera y Arcas, "Communication-efficient learning of deep networks from decentralized data," Proc. 20th Int. Conf. Artif. Intell. Stat. AISTATS 2017, 2017.

[33] X. Sun and N. Ansari, "Dynamic Resource Caching in the IoT Application Layer for Smart Cities," IEEE Internet Things J., vol. 5, no. 2, pp. 606–613, 2018.

[34] Z. Piao, M. Peng, Y. Liu, and M. Daneshmand, "Recent Advances of Edge Cache in Radio Access Networks for Internet of Things: Techniques, Performances, and Challenges," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 1010–1028, 2019.

[35] L. Li, G. Zhao, and R. S. Blum, "A survey of caching techniques in cellular networks: Research issues and challenges in content placement and delivery strategies," IEEE Commun. Surv. Tutorials, vol. 20, no. 3, pp. 1710–1732, 2018.

[36] J. Liu, B. Bai, J. Zhang, and K. B. Letaief, "Content caching at the wireless network edge: A distributed algorithm via belief propagation," 2016 IEEE Int. Conf. Commun. ICC 2016, 2016.

[37] W. Jiang, G. Feng, S. Qin, and Y. Liu, "Multi-Agent Reinforcement Learning Based Cooperative Content Caching for Mobile Edge Networks," IEEE Access, vol. 7, pp. 61856–61867, 2019.

[38] Z. Sang, S. Guo, Q. Wang, and Y. Wang, "GCS: Collaborative video cache management strategy in multi-access edge computing," Ad Hoc Networks, vol. 117, 2021.

[39] S. Dutta and A. Narang, "Dynamic Uncertainty-Based Analytics for Caching Performance Improvements in Mobile Broadband Wireless Networks," Big Data Princ. Paradig., pp. 389–415, 2016.

[40] X. Zhang and Q. Zhu, "Collaborative hierarchical caching over 5G edge computing mobile wireless networks," IEEE Int. Conf. Commun., vol. 2018-May, pp. 1–6, 2018.

[41] D. Ren, X. Gui, K. Zhang, and J. Wu, "Hybrid collaborative caching in mobile edge networks: An analytical approach," Comput. Networks, vol. 158, pp. 1–16, 2019.

[42] T. X. Tran, P. Pandey, A. Hajisami, and D. Pompili, "Collaborative multi-bitrate video caching and processing in Mobile-Edge Computing networks," 2017 13th Annu. Conf. Wirel. On-Demand Netw. Syst. Serv. WONS 2017 - Proc., pp. 165–172, 2017.

[43] Z. Chen, J. Lee, T. Q. S. Quek, and M. Kountouris, "Cooperative Caching and Transmission Design in Cluster-Centric Small Cell Networks," IEEE Trans. Wirel. Commun., vol. 16, no. 5, pp. 3401–3415, 2017.

[44] L. Chen and J. Xu, "Collaborative Service Caching for Edge Computing in Dense Small Cell Networks," pp. 1–30, 2017.

[45] S. Wang, X. Zhang, K. Yang, L. Wang, and W. Wang, "Distributed Edge Caching Scheme Considering the Tradeoff Between the Diversity and Redundancy of Cached Content," 2017.

[46] M. K. Somesula, R. R. Rout, and D. V. L. N. Somayajulu, "Contact duration-aware cooperative cache placement using genetic algorithm for mobile edge networks," Comput. Networks, vol. 193, 2021.

[47] A. Sadeghi, F. Sheikholeslami, and G. B. Giannakis, "Optimal and Scalable Caching for 5G Using Reinforcement Learning of Space-Time Popularities," IEEE J. Sel. Top. Signal Process., vol. 12, no. 1, pp. 180–190, 2018.

[48] Y. Dai, D. Xu, S. Maharjan, G. Qiao, and Y. Zhang, "Artificial Intelligence Empowered Edge Computing and Caching for Internet of Vehicles," IEEE Wirel. Commun., vol. 26, no. 3, pp. 12–18, 2019.

[49] W. Shi et al., "LEAP: Learning-based smart edge with caching and prefetching for adaptive video streaming," Proc. Int. Symp. Qual. Serv. IWQoS 2019, 2019.

[50] I. A. Elgendy, W. Z. Zhang, H. He, B. B. Gupta, and A. A. Abd El-Latif, "Joint computation offloading and task caching for multi-user and multi-task MEC systems: reinforcement learning-based algorithms," Wirel. Networks, vol. 27, no. 3, pp. 2023–2038, 2021.

[51] K. Qi and C. Yang, "Popularity Prediction with Federated Learning for Proactive Caching at Wireless Edge," IEEE Wirel. Commun. Netw. Conf. WCNC, vol. 2020-May, 2020.