Journal of ISMAC (2020)
Vol.02/ No.03
Pages: 160-165
http://irojournals.com/iroismac/
DOI: https://doi.org/10.36548/jismac.2020.3.004

Artificial Intelligence based Edge Computing Framework for Optimization of Mobile Communication

A. Sathesh,
Department of Electronics and Communication Engineering,
Eritrea Institute of Technology,
Eritrea.
Email: Sathesh4you@gmail.com

Abstract: For improving the mobile service quality and acceleration of content delivery, edge computing techniques have been providing optimal solution to bridge the device requirements and cloud capacity by network edges. The advancements of technologies like edge computing and mobile communication has contributed greatly towards these developments. The mobile edge system is enabled with Machine Learning techniques in order to improve the edge system intelligence, optimization of communication, caching and mobile edge computing. For this purpose, a smart framework is developed based on artificial intelligence enabling reduction of unwanted communication load of the system as well as enhancement of applications and optimization of the system dynamically. The models can be trained more accurately using the learning parameters that are exchanged between the edge nodes and the collaborating devices. The adaptivity and cognitive ability of the system is enhanced towards the mobile communication system despite the low learning overhead and helps in attaining a near optimal performance. The opportunities and challenges of smart systems in the near future are also discussed in this paper.

Keywords: Artificial Intelligence; Energy management; Mobile Communication; Edge Computing; Machine Learning;

1. Introduction

In the largely developing mobile networks and heavily congested support networks, the amount of workload imposed is voluminous [1]. This has led to a drastic increase in the mobile communication traffic and computational requirements of the devices and mobile users. The rich multimedia services in terms of quality and quantity has also been increasing rapidly, contributing towards this network congestion [2]. In order to reduce this congestion in the networks, a novel paradigm termed as Mobile Edge Computing (MEC) is introduced. This helps in converting the nearby user equipment (UEs) into storage and computational resources, thereby reducing the network congestion [3]. During data transmission between the cellular devices and the remote cloud computing infrastructures, the long propagation delays that occur are circumvented by the MEC. This enables supporting the Internet of Things (IoT) and mobile application that are latency critical. The pressure of data centers that are cloud based and the load of the mobile communication networks are relieved by the UEs quality-of-experience (QoE) and the mobile network operators (MNOs) quality-of-service (QoS) that are improved by the edge nodes that can handle the content requests and computation of the UEs due to the storage and computational capabilities of the base station [4].

Based on the properties of the MEC, it is insignificant to satisfy the UEs QoE requirements [5]. The computational and communication resources of the edge nodes can be allocated jointly by considering the integrated system of computation and communication on whole that deals with the optimization and decision making issues due to the wireless channel congestion and wireless data transmission requirements that are a major challenge for computation offloading. Game theory, convex optimization and several such schemes and settings are implemented due to the results offered by them by several researchers [6]. Several challenges are faced by the

ISSN: 2582-1369 (online)
Submitted: 22.05.2020
Accepted: 03.07.2020
Published: 08.07.2020

optimization schemes while certain use cases are executed in MEC. They include uncertain inputs, dynamic conditions and temporal isolation. The varying privacy policies and wireless channels cause difficulties in obtaining certain major input factors. This issue is termed as uncertain inputs. The computation system and integrated communication dynamics are not addressed appropriately [7]. Other than Lyapunov optimization, the resource allocation decisions and the current decisions long-term effect are not considered. Only a system snapshot attains the close to optimal or optimal solution in most of the optimization algorithms that are proposed for MEC systems that are highly time-varying. Lack of intelligence is the key issue in optimization of MEC system resource allocation [8].

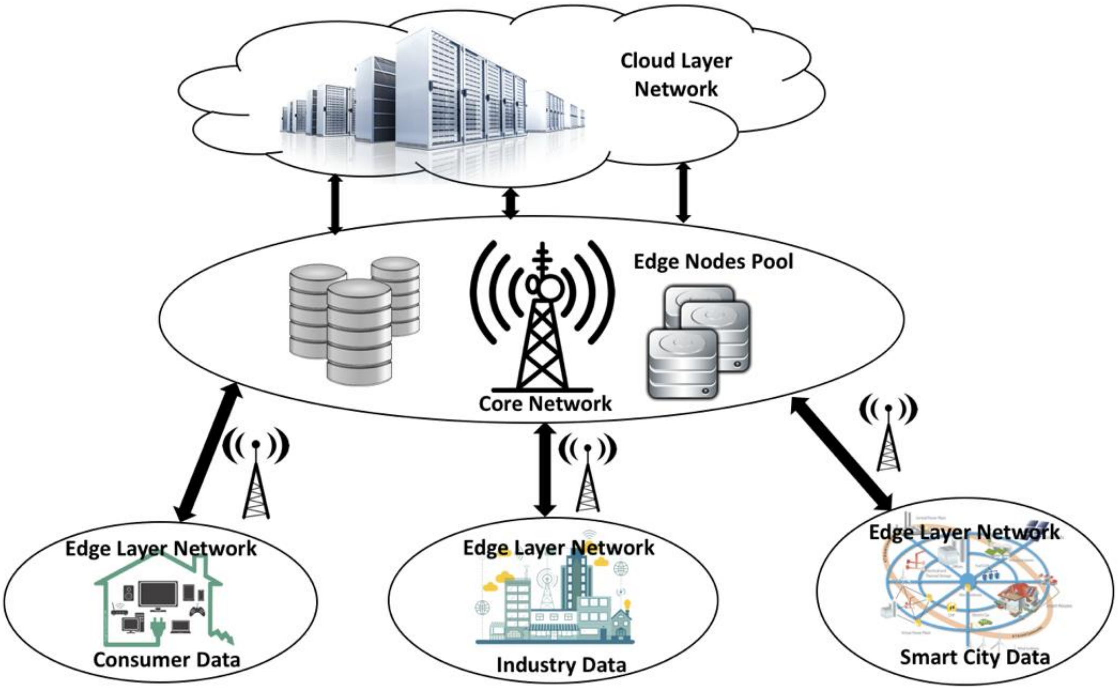


Figure 1: Mobile Edge Architecture with cognitive ability and Artificial Intelligence

2. Edge DRL Scheme

An edge is available between the cloud based and UE artificial intelligence. The huge UEs and cloud can be combined with edge using the cognitive ability and artificial intelligence equipped edge nodes using machine learning framework [9]. The MEC system is balanced and optimized globally by means of the proposed architecture in which the system level AI tasks are supported by the edge nodes. The machine learning integration with AI and edge involves computation offloading and edge caching use cases. The UEs requests content that is cached appropriately and decisions are made by the edge nodes using the DRL agent in MEC systems with edge caching from edge to the cloud. Collaborated edge caching requirements are met by the contents requested by popularity dynamics in the space-time domain [10]. The edge nodes are coordinated by the central cloud server. The local training data is used for updating the edge nodes using DRL agent.

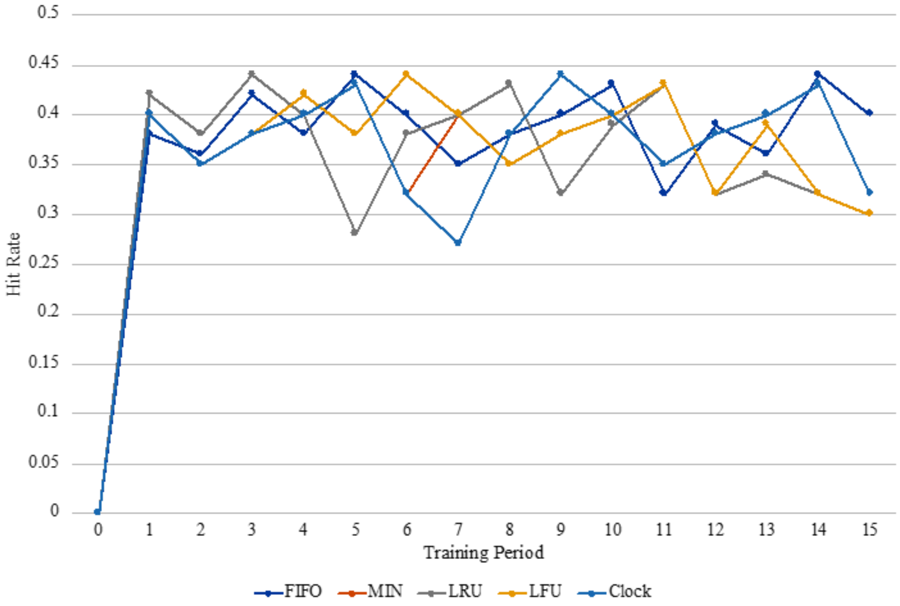
ISSN: 2582-1369 (online)
Submitted: 22.05.2020
Accepted: 03.07.2020
Published: 08.07.2020

$$R_D = E_{MAX} - E_{REAL}$$

The above expression represent the cloud DRL reward in which the highest energy consumption value of the device is represented by E_{MAX} and the At every decision epoch, the device overall energy consumption is given by E_{REAL}. Deep Q-Learning is implemented online after training the DNN in offline mode [11]. It is a five step process as described. For every state, the state-action pair is obtained by the edge DRL with the value of estimated Q function during the start of every decision epoch. Further, execution action is selected using the e-greedy policy for specific device classes. A probability of (1-E) is available for estimation of highest Q-value and choosing the right action accordingly [12]. The action set is used for random selection of action with probability E. The optimal solution is used for the device energy usage management and the selected class is used for obtaining optimal solution using edge DRL. Later, the experience memory is used for storing the class of devices related state transition information which is sampled on observation of next state and immediate reward from the devices. The final step involves obtaining a loss function using the state transitions that are sampled and updating of the weights on DNN for edge DRL agent [13].

3. Proposed Work

The computation offloading decision COD$\in$ {0}$\cup${M}, where M={1,…m) is the wireless channel set representing a single base station with m channels. The local computation and wireless communication is allocated with energy E. These terms decide the control action for making decisions for allocation of computation resources and joint communication by the UE and is represented as (COD, E). This enables the MEC systems to perform computation offloading in a more efficient manner. The UE task execution quality of experience is improved by testing and implementation of various schemes. The major insufficiencies faced during computational offloading due to the MEC wireless communication involves huge training data leading to overloading uplink wireless channel when massive UEs are considered, threat to the privacy of training data, less relevant server side proxy information and so on [14]. Individual training of the UEs often lead to weak computational capability and additional energy consumption during the training process of the UE.



ISSN: 2582-1369 (online)
Submitted: 22.05.2020
Accepted: 03.07.2020
Published: 08.07.2020

Figure 2: Comparison of performance of various edge caching schemes

When massive UEs are involved, obtaining the necessary global data is challenging during the occurrence of task offloading issues in MEC systems with high complexities. The UE joint control action is handled by Deep Q-Leaning and other DRL algorithms. The major challenges of distributed deep learning approaches are categorized as Non-Independent and Identically Distributed (Non-IDD), limited communication, unbalanced, privacy and security. Cognitive capability, robustness and flexibility are the major advantages of the edge AI system using Machine Learning schemes. This concept is future oriented where deep learning model can be implemented in smartphones and mobile devices. Figure 2 represents the performance comparison of various edge caching schemes like click, LFU, LRU, MIN and FIFO.

The MEC system and its training process is boosted using the transfer learning scheme. The UE requests and wireless environment is simulated using this technique. The Deep Reinforcement Learning agent is trained in off-line mode using the simulated environment similar to using a simulated testbed for adjustment and evaluation of the antenna settings. The mobile UEs are distributed with the DRL model that is established. Inference based on the essential level of accuracy is derived from learning which is further derived by training. This process is time consuming. This time consumption can be reduced to a great extent using transfer learning schemes along with training. The training requirement is relieved by the pre-established DRL that enables attaining an appropriate level of accuracy directly after inferring or on using several mini batches. In DRL model, the neural networks can also be adjusted using the DRL established in the UEs when there is a change in the UE pattern or the wireless environment.

4. Results and Discussion

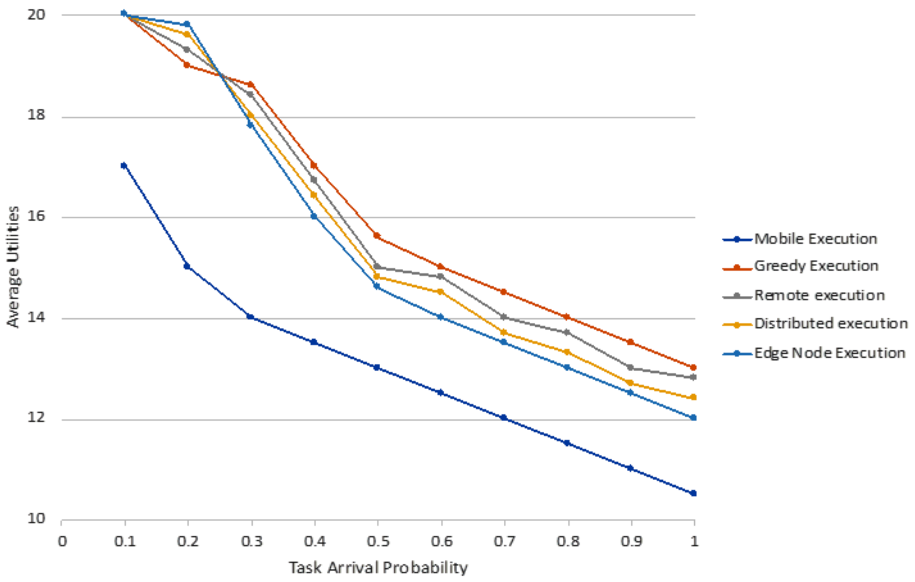


Figure 3: Performance comparison of various offloading schemes

ISSN: 2582-1369 (online)
Submitted: 22.05.2020
Accepted: 03.07.2020
Published: 08.07.2020

Computation offloading as well as edge caching simulation is presented for performance assessment of the edge based artificial intelligence as well as the machine learning scheme respectively. Time epochs are discretization of time horizon for simulation. For the purpose of testing, over a period of one month, the Xender's trace is captured that includes over 10,000 active mobile users and content requests which sum up to around 5 million while the content files conveyed are over 200,000. A Zipf distribution is used for distribution of the content popularity. The six edge nodes cooperation is considered and the UE content requests are generated using the distribution of content popularity based on the Zpif distribution that has been mined for simulation of edge caching. The edge nodes are coordinated by the central server which is the MNOs cloud server in edge caching.

The caching decision is made by the local DRL agent on reception of UW request from the edge node. Further, the training is performed using the reward obtained for this action. Ten wireless channels may be obtained by dividing an edge node of 5MHz bandwidth for computation offloading of the MEC system with Machine learning and DRL capabilities. The six levels of wireless channel quality are quantified by a common finite set of edge node and UEs. Bernoulli distribution is used for analysis of the generated task count throughout simulation for the time epochs. The centralized DLR agent is constructed for comparison assuming that all reinforcement learning based data can be received by the system.

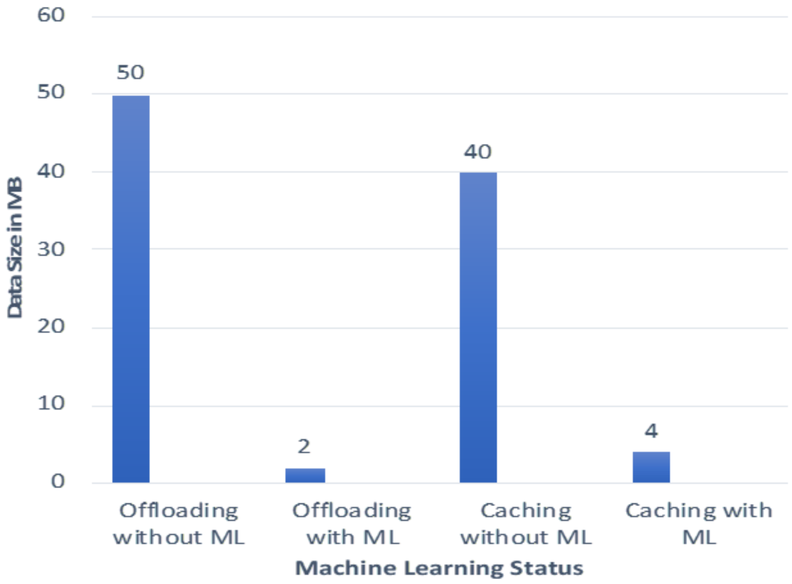


Figure 4: Transmission Cost comparison with respect to the machine learning status

Different settings are used for testing the computation offloading and edge caching schemes are represented in Figure 4 to analyze the framework of AI-supported mobile edge system with cognitive ability and its performance. The Deep Reinforcement Learning agents and their capabilities are exhibited using a specific number of UEs and edge nodes. This architecture is used for computation offloading and edge caching performance analysis using Double Deep Q Learning and machine learning scheme. Greedy execution, edge node execution, mobile execution, remote execution as well as distributed execution are the baseline computational offloading schemes that are compared for efficiency. Local processing of computation tasks is enabled. Till the

DLR agent is trained for error free operation the total wireless transmission statistics are gathered in both computation offloading as well as edge caching.

5. Conclusion

An Artificial Intelligence based edge computing framework for optimization of mobile communication is discussed in this paper with federated and deep reinforcement based machine learning schemes. Caching is also optimized along with the mobile edge communication. Different caching and offloading schemes are analyzed for edge computing and mobile communication. Further, various scenarios are analyzed for offloading and caching, with and without machine learning techniques for experimentation. Evaluation of the proposed is performed and near-optimal performance is attained. Future work is to be focused on collaboration and fine-grained structure of mobile devices and edge nodes over scheduling tasks for computational task learning optimization.

References

[1] Zhou, Z., Chen, X., Li, E., Zeng, L., Luo, K., & Zhang, J. (2019). Edge intelligence: Paving the last mile of artificial intelligence with edge computing. Proceedings of the IEEE, 107(8), 1738-1762.

[2] Cao, B., Zhang, L., Li, Y., Feng, D., & Cao, W. (2019). Intelligent offloading in multi-access edge computing: A state-of-the-art review and framework. IEEE Communications Magazine, 57(3), 56-62.

[3] Li, X., Li, D., Wan, J., Liu, C., & Imran, M. (2018). Adaptive transmission optimization in SDN-based industrial Internet of Things with edge computing. IEEE Internet of Things Journal, 5(3), 1351-1360.

[4] Sodhro, A. H., Luo, Z., Sangaiah, A. K., & Baik, S. W. (2019). Mobile edge computing based QoS optimization in medical healthcare applications. International Journal of Information Management, 45, 308-318.

[5] Wang, X., Han, Y., Leung, V. C., Niyato, D., Yan, X., & Chen, X. (2020). Convergence of edge computing and deep learning: A comprehensive survey. IEEE Communications Surveys & Tutorials, 22(2), 869-904.

[6] Zhang, K., Leng, S., He, Y., Maharjan, S., & Zhang, Y. (2018). Mobile edge computing and networking for green and low-latency Internet of Things. IEEE Communications Magazine, 56(5), 39-45.

[7] Li, H., Ota, K., & Dong, M. (2018). Learning IoT in edge: Deep learning for the Internet of Things with edge computing. IEEE network, 32(1), 96-101.

[8] Deng, S., Zhao, H., Fang, W., Yin, J., Dustdar, S., & Zomaya, A. Y. (2020). Edge intelligence: the confluence of edge computing and artificial intelligence. IEEE Internet of Things Journal.

[9] Li, X., Wan, J., Dai, H. N., Imran, M., Xia, M., & Celesti, A. (2019). A hybrid computing solution and resource scheduling strategy for edge computing in smart manufacturing. IEEE Transactions on Industrial Informatics, 15(7), 4225-4234.

[10] Pandian, A. P. (2019). Enhanced edge model for big data in the internet of things based applications. Journal of trends in Computer Science and Smart technology (TCSST), 1(01), 63-73.

[11] Bhalaji, N. (2020). Efficient and Secure Data Utilization in Mobile Edge Computing By Data Replication. Journal of ISMAC, 2(01), 1-12.

[12] Sivaganesan, D. (2019). DESIGN AND DEVELOPMENT AI-ENABLED EDGE COMPUTING FOR INTELLIGENT-IOT APPLICATIONS. Journal of trends in Computer Science and Smart technology (TCSST), 1(02), 84-94.

[13] Chen, J. I. Z. COMPUTATIONAL OFFLOADING FOR PERFORMANCE IMPROVEMENT AND ENERGY SAVING IN MOBILE DEVICES.

[14] Raj, J. S. (2019). A comprehensive survey on the computational intelligence techniques and its applications. Journal of ISMAC, 1(03), 147-159.

ISSN: 2582-1369 (online)
Submitted: 22.05.2020
Accepted: 03.07.2020
Published: 08.07.2020