

Deep Reinforcement Learning for Interactive Mobile Experiences

Dheeraj Vaddepally

dheeraj.vaddepally@gmail.com

Abstract

The emergence of the mobile applications that are driven by machine learning (ML) has brought a breakthrough in the world of healthcare, e-commerce, and entertainment with its personalized and smart user experiences. Despite the wide application of ML pipelines in mobile environments, the security and privacy issues that could be raised have become an issue of high concern on a global scale. The storage and processing of data, combined with the deployment of advanced ML models, expose vulnerabilities to risks such as data leakage, adversarial attacks, and unauthorized access. These threats not only compromise user trust but also incur applications with financial and brand damage.

The present article deals with the important problems of the safety of mobile ML pipelines, namely adversarial manipulations, insecure storage, and transmitting of data. The vital techniques that can help to work out those problems are described along with the most popular ones, namely secure model storage based on encryption, federated learning for data transfer reduction, and model robustness tactics that will increase adversarial attacks defense. Besides that, new technologies like homomorphic encryption and blockchain are also discussed as ways to secure model updates. Consequently, this paper is involved in the technical and practical levels of the project, and it is meant to show the importance of building secure and privacy-preserving ML pipelines that do not only keep data for user trust but also maintain efficiency. It serves the purpose of opening the way to moving to secure mobile ML solutions in the time of intelligent mobile systems that have started to be of great interest.

Keywords: Reinforcement learning, DRL, Artificial Intelligence, Machine learning, Features, Mobile Applications

I. INTRODUCTION

The sudden evolution of mobile applications in recent years has successfully changed the approach to technology users' interaction, as people now require an experience that is of more functionality but at the same time also of intuition and invitation. AI technology has played a significant role for the last few years in resonating not only interaction but even more in impacting very much the interactivity. AI subset called Reinforcement Learning is presented as a learning technique where the system interacts with the environment and learns from its behavior with its application to be learned [1].

Reinforcement Learning is the concept of learning through trial and error, which works by the fact that the agent learns the right steps by receiving either a positive reward or a negative one. The RL has been further reinforced through Deep Learning (DL) into Deep Reinforcement Learning (DRL), it is combining the decision-making capabilities of RL with the representation learning power of neural networks. DRL technology has been successfully applied to problem spaces such as gaming, robotics, and autonomous vehicles. Its innovation to adapt on the fly everything a user wants and likes in the mobile world has now become a reality [1] [2]. Even though there is a good prospect of DRL to have a huge impact on real-time interactivity and personalization in mobile environments, as usual, there are lots of difficult and challenging problems that need to be solved at

first. The limited computing capabilities related to CPU power, memory, and energy of a mobile device are the constraint to tackle the resource-demand upside down with heavy computation needed at different instances [2].

The issues of developing a mechanism that can accurately identify user engagement and selecting the best way to function in a volatile and dynamic environment arise as problems. One more major obstacle is the fact of actually applying DRL on mobile devices. Mobile devices need on-device training and inference solutions which is balance of power, efficiency, and privateness. Indeed, if we don't sort out these challenges, the possibility of DRL being an attractive tool for enforcing user-centric mobile technologies is still untapped.

This paper becomes the research that is oriented to the application of DRL techniques in the field of creating interactive mobile experiences that put the user at the center of their decision-making. [3]. It is dynamic on the importance of reward design when it comes to converting user actions into actionable feedback for DRL agents and the optimization of policies for decision-making improvement. Joining that, the journal analyzes categorizes additional issues to be considered when deploying DRL models on mobile phones, such as model compression, federated learning, and edge AI frameworks [4].

II. LITERATURE REVIEW

A. Evolution of Reinforcement Learning and Its Mobile Applications

Reinforcement Learning (RL) is a machine learning paradigm that is used to make interactions with an environment based on rewards or penalties, which an agent character learns by ad hoc algorithm. The use of neural networks with deep RL minimized these limitations by creating the possibility of implementing value functions, policies, and rewards. This method has demonstrated its effectiveness in the resolution of multidimensional problems such as playing games like Go, Dota 2, and chess where it beat human performance in the long run [3].

When it comes to DRL, it is very popular on the mobile platforms these days as it provides the capability of doing it. It helps in personalizing the user experiences in real-time is through this approach. For example, DRL technology that is used in gaming is also a good choice because it produces games for the players who can be engaged by it, it can be used in recommendation systems to tailor content suggestions based on user behavior, and it helps with user interfaces by optimizing layouts and features based on contextual user data. All of these are features that illustrate the potential that DRL has to change from a static and predefined to a more dynamic and user-centric mobile environment [5].

B. Reward Design and Policy Optimization

Composing a reward system is the most crucial element of an RL system, as it is responsible for the feedback the agent gets for its decisions. For a mobile application, designing reward mechanisms that not only tell users what their clicks, dwell times, and navigation patterns are but also reward them for it is the major role of the agent. It is important to know how to give immediate rewards and how to balance those versus long-term gratifications by which we rule out the option of user's boredom on the other side of the spectrum [6].

Policy optimization is a process of enhancement of the agent's capacity to make the best decision by modifying its policy, which is a set of rules from states to actions. The most popular and commonly used techniques for DRL are Deep Q-Learning and Proximal Policy Optimization (PPO). Deep Q-Learning includes approximating Q-value function through neural networks that enable the agent to take actions that will bring it the maximum cumulative rewards. PPO is a policy gradient method that centers on controlling the changes in the policies and it is comparatively stable and efficient due to the last policy's influence. Such methods are pivotal for the practical application of DRL agents in complex and dynamic environments [5].

C. On-device training and inference

Mobile integration of reinforcement learning models is complicated due to limitations on compute power, memory capacity, and battery longevity.

On the user's device, the update of the models directly via on-device training call for lightweight algorithms along with an efficient optimization technique. In the case of on-device inference, trained models are to be used in real time to make the right decisions while the primary goal is to minimize both latency and energy [7].

Libraries including TensorFlow Lite, PyTorch Mobile, and Core ML have a key function in making DRL on device become a technology that is widely used. A crucial among these frameworks is the model compression capabilities, like quantization and pruning that focus on the size and computational requirements of the DRL model, without degrading its performance greatly. Beyond that, the federated learning methods also serve as the medium for decentralized training on multiple devices, and they not only invest in data privacy but also provide distributed computing resources [5].

DRL technique can be applied on the phone by which then data transfer technology connects the mobile app straight to the mobile app on the other end of the digital line of communication to enable the user to get the best possible service experience. It turns out that new technologies have plunged mobile AI research into a lot of excitement while finding the best solutions for the users [6] [7].

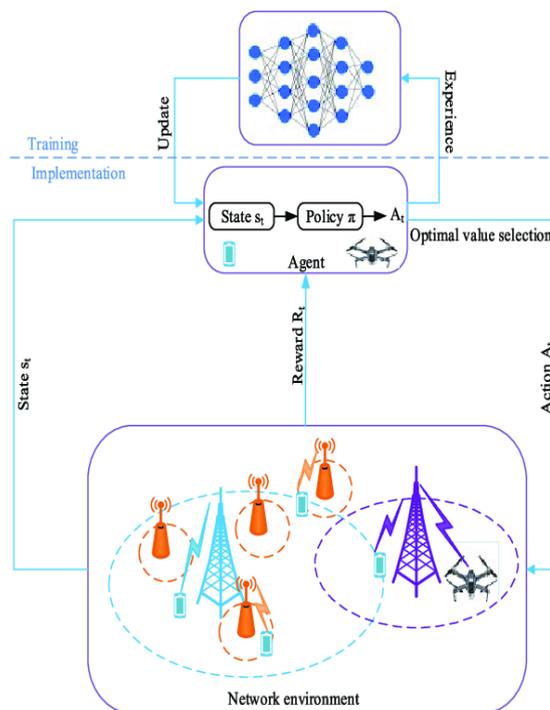


Fig 1. Deep reinforcement learning in mobile networks [7].

D. Reward Design and Policy Optimization for User Engagement

1. Designing Effective Reward Mechanisms

Reward design is the main aspect of reinforcement learning (RL) which is the exertion of every specific stimulus that is able to drive the learning procedure. In the mobile apps, Deep Reinforcement Learning (DRL) demands that the provision of better reward incentives that result in the interaction between the individuals gets converted into practical feedback for the users [8].

- Mapping User Actions to Rewards:

In mobile environments, user actions such as clicks, navigation patterns, dwell time, and retention rates can be mapped to reward signals. For example:

Dwell time: More use of the application leads to increased dwell time, which should, in turn, mean that the user gets more rewards [9].

The likes, shares, or the percentage of tasks successfully completed could also be taken as engagement metrics of user satisfaction [10].

Negative Feedback: Quickest exits or uninstalls could act as a punishment for actions that would eventually cause the worst experience of a user.

By aligning such actions with the application-specific goals, DRL agents can effectively learn to optimize for user satisfaction and engagement.

- **Balancing Immediate vs. Long-Term Rewards:**

Rewards designing has been one of the hard things is to maintain the balance between the user engagement in the short term and long term. Immediate rewards involve short-term optimization (e.g., getting the user to click on a recommendation). However, these may not always result in sustained engagement, and especially if they tend to shallow or manipulative designs (e.g., clickbait) the user might get alienated. Though, in the other case, a streaming service might still give the content an agent should recommend a show that a user clicks on the platform, provided, in the long term, the user actually completed the series or even extended the subscription. This worker has not only received the customer's short-term preference but kept it in the long run by using both the short-term reward and long-term reward [11].

2. Policy Optimization Techniques

Optimization is enhancing the decision-making policy of the agent so that it maximizes the cumulated rewards during the interaction process. Generally, this refers to learning how to pick its best response to any actions within a user session given its observed state and this user's stated preference [6].

- **Model-Free Methods:**

Model-free approaches, for example, Deep Q-Learning and Proximal Policy Optimization (PPO), are extremely popular in mobile environments as they are easy to implement and highly effective.

Deep Q-Learning: Here, the Q-value function is approximated by a neural network that predicts the reward corresponding to a state and an action. For instance, the difficulty level of a mobile game application can be varied dynamically to coincide with that of a player skill using Deep Q-Learning so that the player engages in the game.

Proximal Policy Optimization (PPO)

PPO limits large-scale policy variations throughout the optimization process and reduces instability in learning. This model proves particularly beneficial when dealing with dynamically changing mobile application environments, including virtual assistants with the need to respond to preferences [12] [9].

Model-based DRL creates a predictive model of the environment, such that the agent can use it to "rehearse" likely scenarios before an action is taken. Therefore, this strategy eliminates most of the trial-and-error nature in real-world environments. For example, within an online shopping application, the model-based agent would forecast the consequences of suggesting specific products to the actual user purchases as well, thus enhancing the decisions themselves [7].

3. Dynamic Environments Strategies

The DRL system must be made more versatile so it can adapt with user preferences and changing contexts. Therefore, the approach may include some of the following:

Context-Aware Learning: It is defined as using parameters, including location, device, and time of day, to further define the recommendations [13].

Meta-Learning: Training agents to swiftly acquire new competencies or understand human behavior with limited data. For instance:

- **Mobile Games:** DRL improves mobile games by improving the gameplay experience in real-time, hence sustaining player engagement.

- The game Honor of Kings, developed by Tencent, employs Deep Reinforcement Learning to enhance matching and game complexity. By monitoring player activity, game levels and level complexity may be changed. This keeps the gamers in the game and interested.

- Virtual Assistants: DRL had paved the road to the generation of virtual assistants, such as Siri, Alexa, and Google Assistant [14] [12].

Such a system applies DRL to learn the intent of the user and develops their interaction through feedbacks. For instance, if a user never stops reminding him/her of certain tasks, an assistant learns to proactively suggest the same kind of activity, thus enhancing relevance and usability.

4. E-commerce platforms

For instance, the leading e-commerce websites such as Amazon and Alibaba are utilizing DRL in recommendation systems. DRL agents fine-tune product recommendations by maximizing click-through rates and conversion according to user behavior. For example, an agent may rank items according to the browsing history of a user while maintaining the balance of the inventory and the profit margin [10].

5. Streaming services

For example, Netflix and Spotify use DRL to select the content based on the preference of users. The reward mechanisms are optimized to the streaming of watch time, playlist completions, and a rating system of a user to keep them on the platform. DRL agents change recommending lists of products in accord with dynamically changing demands, thus, giving a user-tailored experience [3].

The design of rewards and policy optimization are critical for driving DRL applications to a higher level of user engagement realization in mobile apps. The use of well-designed rewarding schemes is one of the main aspects involved in strategizing of core user appeals and business perspectives, while on the other hand, advance policy optimization techniques provide adaptable learning in ever-changing ambient settings.

Overcoming these challenges can be attained by mapping user actions to rewards, planning objectives for the short and long durations, and by using the state-of-the-art DRL methods, in this way, mobile apps can be interactive and personalized to a whole new level. Some excellent examples, in gaming, virtual assistants, and e-commerce, have been embedded with DRL technology to prove its empowerment for such mobile applications [6].

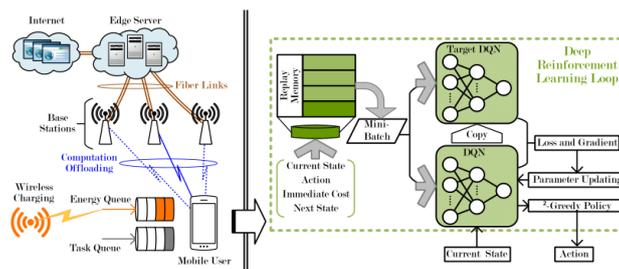


Fig 2. Illustration of deep reinforcement learning for mobile-edge computing performance optimization [15].

III. PRACTICAL CONSIDERATIONS FOR ON-DEVICE TRAINING OR INFERENCE

The problem of on-device training and inference for DRL is unique due to the limitations of mobile devices, but at the same time, they provide a set of solutions that could make it possible to achieve efficient, real-time, user-centric applications. The application is designed to protect the privacy of users in mobile environments while also leveraging machine learning to keep the program's speed and efficiency at a high level [15].

A. Limitations

1. Limitations imposed by mobile devices in terms of computation:

The processing capabilities of mobile gadgets, particularly smartphones and tablets, are drastically diminished in comparison to those of traditional computers.

The limitation is due to less memory and processing power with a much smaller capacity for storage. Most DRL models, which demand considerable computational resources in learning and optimizing, conflict with the mobile hardware's limitations, thus causing the training to take longer, resulting in low performance, or failing to execute on the device. Mobile devices might also not be powerful enough to execute complex DRL models in real-time, thereby resulting in latency and a much less interactive user experience [7].

2. Energy Efficiency Issues:

Applications of DRL that require constant updating or even immediate decision-making may consume much more energy. Mobile devices run on batteries; long-running computationally expensive tasks like model training or complex inference will significantly deplete the battery. In DRL, which is a continuous learning and decision-making process based on user interactions, it is important to identify ways to optimize energy consumption. With mobile devices having very short battery life, they can ill afford the heavy computation cost associated with running DRL models unless such models are optimized to consume as little power as possible [12].

B. Solutions

1. Model Compression Techniques

It involves reducing the size of a pre-trained model such that its performance is not significantly compromised. Common techniques for model compression are:

Quantization: The weights and activations of the model could be reduced to much less precision, for instance from 32-bit floating-point numbers to 8-bit integers, causing an enormous reduction in size and its implication on the computation load to the mobile device. By doing this, quantization allows models to run more efficiently on mobile devices with limited computational resources [2] [3].

Pruning: This technique involves removing redundant or unimportant parameters from the model. By eliminating neurons or weights that have little effect on the model's performance, pruning reduces the size and complexity of the model, which translates into faster inference and lower memory usage [5].

Both of these approaches may make it feasible for a mobile device to execute DRL models in real time with a very minor penalty in terms of performance, energy consumption, and memory usage.

2. Federated learning for distributed training

One of the most promising techniques in overcoming limitations in mobile devices is the federated learning technique, in which the training data remains on the user's device while only the model updates are transferred to a central server. Federated learning enables the model to be trained collaboratively across multiple devices without requiring users' data to leave their devices, ensuring privacy and data security [7].

In DRL, federated learning can enable the training of sophisticated models while keeping computational demands distributed. Since the training data is located on each device, each device may improve the model by updating its local model with its interaction data. The updates are then aggregated on a server to improve the global model, enabling the system to exploit data from a large number of devices with a guarantee on privacy. This improves DRL models without real-time training in the server, thus improving mobile device compute constraints [7].

3. Inference Pipeline Optimization using Edge AI Frameworks:

They facilitate real-time inference of mobile Edge AI frameworks based on DRL.

They have been designed for resource-constrained environments and optimized tools to take AI models onto mobile devices.

Core ML (Apple): Apple's framework to integrate machine learning models into an iOS device is Core ML. It supports deep learning models among many others and optimizes for on-device inference to ensure that the DRL application runs efficiently on an iPhone or an iPad. It automatically converts the model to an optimized

format for mobile devices, which greatly enhances performance while cutting down on memory and processing needs [7].

ONNX Runtime: Open Neural Network Exchange (ONNX) is an open-source framework aimed to facilitate the cross-platform deployment of deep learning models. ONNX Runtime optimizes models for edge devices, which supports a range of hardware accelerators, including CPUs, GPUs, and other specialized processors. In this way, DRL models can be executed efficiently on Android and iOS devices with high performance [8].

Both Core ML and ONNX Runtime allow the optimization of DRL applications on mobile devices, where decision-making happens with low latency, thereby providing an excellent user experience, even with highly complex DRL models.

4. Security and Privacy

The data privacy and security must be guaranteed with the rising penetration of AI in mobile devices. Sensitive user data are required for on-device training, so they need to be secured. The techniques that ensure data protection both during the training and inference processes are:

- Data privacy in on-device training

The on-device training ensures no one sends any kind of user data to the remote servers, therefore, the leakage chance is very low; however, there will be leakage chances of sensitive data if it has not been dealt with appropriately. Differential privacy can be used with the upgrade, which assures confidentiality of data, ensuring the upgraded model cannot decode any data even a single point. Therefore, user confidentiality will not be affected [12].

- Safe Updates of Models and Data Protection of Users

In federated learning, the utmost importance lies in encrypting the updates of the model fed to the devices and servers. The aggregation of model updates using SMPC ensures that the model updates are aggregated in a privacy-preserving manner [7].

The encryption can be such that computations are allowed on encrypted data so model updates can be performed without risk of leakage of underlying user data. To that end, one may employ technologies of secure model verification to check for authenticity in updates. These technologies make it clear that the updates originate from the authorized devices, which in turn stops the enemies from injecting bad updates into the global model [9].

Mobile applications that allow on-device training and inference for DRL face a number of problems that are not encountered by other applications. In those cases, the main problems include computational limitations, energy efficiency concerns, and privacy issues. Despite that, by the application of techniques such as model compression (quantization and pruning), federated learning, and optimized inference pipelines through edge AI frameworks, those obstacles can be dealt with. Through techniques like differential privacy, homomorphic encryption, and secure multi-party computation, data privacy and secure model updates are ensured to enhance integrity and security of mobile DRL applications. From the concept of DRL being used in mobile devices, these devices enable a more personal, more real-time, and more user-centric experience without sacrificing the performance, privacy, or security [2].

IV. FUTURE SCOPE

The area of Deep Reinforcement Learning (DRL) in the field of interactive mobile experiences has been changing very fast; it has been strongly influenced by exciting advancements and challenges which are shaping its future. In this regard, the section has been mentioned outlining that new trends, ethical and technical issues, and a potential prospect that can drive the growth of the innovation domain [9].

A. Emerging Trends

1. Advanced DRL Algorithms for Cross-Platform Optimization

With the rise of the interface of mobile devices, the focus shifts mainly to DRL algorithms and the necessity of the common operation on different platforms. The evolution of the DRL systems of the future will undoubtedly be the result of the implementation of hierarchical reinforcement learning and multi-agent learning in the process of user experience interaction across different types of devices such as smartphones, tablets, and wearables. Cross-platform optimization improves user experience by making standardized and personalized content available on all devices used by customers [14] [2].

2. Integration of Multi-Modal Data in Reward Design

The subsequent generation of DRL models will consider multi-modal data such as sensor inputs, voice commands, gestures, and contextual data like location and time of day. The use of all these diverse data sources will help a DRL model to understand user preferences better, thereby configuring the experience for each individual's need. This would result in increased engagement and satisfaction levels [15].

3. Advanced On-Device Capabilities

Developments in hardware accelerators, in particular, of any neural processing unit and edge computing frameworks, enable runtimes for complicated DRL algorithm on mobile equipment. Such technology will help in real-time decision-making and richer interactions without relying heavily on cloud infrastructure [7].

B. Ethical and Technical Challenges

1. Biases in Reward Signals

Artificial Intelligence, said to be one of the most significant technologies in the world today, includes deep reinforcement learning, or simply put, and is getting coupled with AI. The challenges that are currently posed to humans are winding up unsolved.

One of the critical challenges in DRL is a biased reward mechanism, which can cause the machine to learn by chance and may lead to the inadvertent promotion of undesirable behaviors. For example, a powerful signal which rewards the retention time on an app can enforce addictive behavior and as a result hinder user well-being. Overcoming these biases will involve the construction of ethical customer-centric reward systems in which greater emphasis is placed on productivity or mental health [7].

2. Balancing Personalization with User Autonomy and Privacy:

Although DRL's major advantage is personalization, personalized solutions may cause people to stop thinking for themselves and instead do what the system suggests. A truly personal and user-centered solution to deliver that same kind of unique customer experience, the system should not only be transparent but also configurable enough for the users to keep, adjust, or even abandon any personalized experience. A big part of the issue can be privacy. Generally, the participation of sensitive data in on-device training has been a massive privacy issue. Hence the future DRL applications have to embrace the new privacy-preserving techniques along with the old user privacy agreements to sustain user confidence [8] [1].

3. Possible Uses in Internet of Things Ecosystems and Smart Devices

DRL has the possibility to radically revolutionize Internet of Things environments and intelligent devices such as linked machines and intelligent homes. In order to improve the energy efficiency of a smart home, different technologies that were created by DRL need to be equipped to autonomously adapt their performances to align with the choices of the user.

The implementation of AR and VR will assist DRL in discovering novel applications for creating immersive and adaptable environments. The utilization of Deep Reinforcement Learning in the development of interactive AR/VR systems will dynamically adjust according to user behavior, exemplified by personalized virtual shopping experiences and adaptive training simulations [12].

C. Expansion in the Healthcare Industry and Making Available

The delivery of healthcare would improve due to the use of DRL-driven mobile applications, which can be more suited for health monitoring, rehabilitation, and assistive technology for a disabled person. Wearable devices, which can be upgraded to carry DRL, can then provide user-specific requirements that allow one to get instant feedback and suggestions [11].

This avenue creates an opportunity through which DRL can create an adaptation of immersive and interactive environments by using the technology of AR and VR. Deep Reinforcement Learning will be used in the construction of dynamic interactive AR and VR systems that adjust based on user activity, such as customized virtual shopping experiences and adaptive training simulations.

- Expansion into Healthcare and Access

DRL-based mobile applications could help in tailoring individual health monitoring, rehabilitation, and assistive technology for an individual with a disability. For example, the wearable device built on DRL can adjust their functionality to tailor it to one user's specific needs, then provide instant feedback and recommend corrective action [5].

- AR/VR Experiences

The emergence of augmented reality (AR) and virtual reality (VR) presents novel opportunities for DRL in developing immersive, adaptable environments. DRL may be employed to develop interactive AR/VR systems that adapt in real-time to user activity, including customized virtual retail experiences or responsive training simulations [2].

- Expansion into Healthcare and Accessibility

DRL-based mobile applications can enhance healthcare by providing personalized health monitoring, rehabilitation programs, and assistive technology for people with impairments. Wearable devices based on DRL may adapt to users' specific needs and provide real-time feedback and recommendations [15].

V. CONCLUSION

Deep Reinforcement Learning (DRL) is undoubtedly a revolutionary technology that is ushering in a totally new era in which consequent changes in the interactive mobile services field would result in dynamic, personalized, and customer-oriented applications. By using DRL, power users get to build instruments that learn and fine-tune themselves with user behavior at the prompt time, which in turn results in them offering the users a more interesting and understandably useful interface. Momentarily, the contrary set of the custom experience is also possible, of course, via the option to inquire the user about opting for mild reactions or to inquire for inquiries that could be possibly explored in the far future. The ability to orchestrate user interactivity in the form of well-delivered rewards and intelligent policy design that ensures lean learning has given rise to creative applications in gaming, e-commerce, education, and healthcare [4].

The effective reward is the essential component of the smooth operation of DRL systems. Linking the actions that users take with relevant rewards in some cases, rather for immediate enjoyment or more strategic long-term engagement is the fundamental rule of thumb for system setting. Equally critical is policy optimization, where advanced DRL algorithms like Proximal Policy Optimization (PPO) and Deep Q-Learning ensure efficient learning and adaptation in dynamic mobile environments.

DRL's future is becoming more optimistic. The obstacles that environments solve the use of DRL are largely based on the nature of mobile devices. The systems are not standalone but are enabling the user by applying the inference generated from periodic updates of the cloud AI model to the prediction. The user should not be aware of security risks, and the battery power consumption should be minimized by the mobile device. One of the growth areas that researchers who focus on such implementations of DRL that allow it to be carried out on the devices of the users will direct is the struggle for making the implementations more flexible as well as easier to scale [6] [7].

Furthermore, the integration of multi-modal data sources and optimization for cross-platforms can lead to optimizing experiences on mobiles to a higher level. Through the mitigation of these constraints and the taking of the train of coming prospects, DRL may enormously streamline the mobile interface as well as inspire the development of genuine trust and ethical propriety of its applications.

Summing up the ideas, the DRL algorithm is one of the most transformative aspects in modern technology yet offers a great opportunity for innovation through the use of this technology to guide businesses to a point where they are able to prioritize issues that the users might come up with. The technology can, however, continue to greatly improve user engagement and experience while its development is grounded in ethical principles and a practical usability premise for it to become its full potential in a responsible way.

REFERENCES

- [1] M. Abdoos and A. L. C. Bazzan, "Hierarchical traffic signal optimization using reinforcement learning and traffic prediction with long-short term memory," *Expert Systems with Applications*, vol. 171, p. 114580, Jun. 2021.
- [2] B. Abdulhai, R. Pringle, and G. J. Karakoulas, "Reinforcement learning for true adaptive traffic signal control," *Journal of Transportation Engineering*, vol. 129, no. 3, pp. 278–285, May 2003, doi: 10.1061/(ASCE)0733-947X(2003)129:3(278).
- [3] S. Anokye, D. Ayepah-Mensah, A. M. Seid, G. O. Boateng, and G. Sun, "Deep reinforcement learning-based mobility-aware UAV content caching and placement in mobile edge networks," *IEEE Systems Journal*, vol. 16, no. 1, pp. 275–286, Mar. 2022, doi: 10.1109/JSYST.2021.3082837.
- [4] J. Chen, Y. Yang, F. Liu, X. Luo, and L. Hu, "Deep reinforcement learning in maximum entropy framework with automatic adjustment of mixed temperature parameters for path planning," in *Proc. 7th Int. Conf. Robotics Control and Automation (ICRCA)*, 2023, pp. 78–82.
- [5] Z. Dai, H. Wang, C. H. Liu, R. Han, J. Tang, and G. Wang, "Mobile crowdsensing for data freshness: A deep reinforcement learning approach," in *Proc. IEEE INFOCOM*, May 2021, pp. 1–10, doi: 10.1109/INFOCOM42981.2021.9488791.
- [6] R. Singh and L. Xu, "A review of deep reinforcement learning algorithms for mobile robot path planning," *Vehicles*, vol. 5, no. 4, pp. 1423–1451, 2023.
- [7] W. Tashan, I. Shayea, S. Aldirmaz-Çolak, O. A. Aziz, A. Alhammadi, and Y. I. Daradkeh, "Advanced mobility robustness optimization models in future mobile networks based on machine learning solutions," *IEEE Access*, vol. 10, pp. 111134–111152, 2022, doi: 10.1109/ACCESS.2022.3215684.
- [8] K. Zhu and Z. Tang, "Deep reinforcement learning-based mobile robot navigation: A review," *Tsinghua Science and Technology*, vol. 26, no. 5, pp. 674–691, 2021.
- [9] H. Zhou, J. Kong, L. Xiao, X. Liu, and V. C. M. Leung, "Deep reinforcement learning for energy-efficient computation offloading in mobile-edge computing," *IEEE Internet of Things Journal*, vol. 9, no. 2, pp. 1517–1530, 2022, doi: 10.1109/JIOT.2021.3091142.
- [10] T. Wang, J. Cao, and A. Hussain, "Adaptive traffic signal control for large-scale scenario with cooperative group-based multi-agent reinforcement learning," *Transportation Research Part C: Emerging Technologies*, vol. 125, p. 103046, 2021.
- [11] K. Arulkumaran, M. P. Deisenroth, M. Brundage, and A. A. Bharath, "A brief survey of deep reinforcement learning," *arXiv preprint arXiv:1708.05866*, 2017.

- [12] L. Liao, Y. Yan, F. Zhang, and Z. Wang, “Online computation offloading with double reinforcement learning algorithm in mobile edge computing,” *Journal of Parallel and Distributed Computing*, vol. 171, pp. 28–39, 2023.
- [13] E. P. Garaffa, B. Manzoni, K. Atkinson, A. Argyros, and F. E. Pont, “Reinforcement learning for mobile robotics exploration: A survey,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, no. 8, pp. 3796–3810, 2021.
- [14] M. Goudarzi, M. S. Palaniswami, and R. Buyya, “A distributed deep reinforcement learning technique for application placement in edge and fog computing environments,” *IEEE Transactions on Mobile Computing*, vol. 22, no. 5, pp. 2491–2505, May 2023.
- [15] M. S. Elbamby, C. Perfecto, C.-F. Liu, J. Park, S. Samarakoon, X. Chen, and M. Bennis, “Wireless edge computing with latency and reliability guarantees,” *Proceedings of the IEEE*, vol. 107, no. 8, pp. 1717–1737, Aug. 2019, doi: 10.1109/JPROC.2019.2917084.