Jurnal Teknik Informatika C.I.T Medicom, 15 (3) (2023) 118-125

Published by: Institute of Computer Science (IOCS)

Jurnal Teknik Informatika C.I.T Medicom

Journal homepage: www.medikom.iocspublisher.org

Privacy-Preserving machine learning in edge computing environments

Deni Kurniawan [1], Dedi Triyanto [2], Mochamad Wahyudi [3], and Lise Pujiastuti [4]

[1,2] Program Studi Sistem Informasi, Universitas Bina Sarana Infotmatika, Jakarta, DKI Jakarta, Indonesia

[3] Program Studi Ilmu Komputer, Universitas Bina Sarana Infotmatika, Jakarta, DKI Jakarta, Indonesia

[4] Program Studi Sistem Informasi, STMIK Antar Bangsa, Tangerang, Banten, Indonesia

Article Info

Article history

Received : Apr 30, 2023
Revised : May 18, 2023
Accepted : Jul 3, 2023

Keywords:

Data Privacy;
Decentralized Computing;
Edge Computing;
Federated Learning;
Privacy-Preserving Machine Learning.

Abstract

Edge computing has transformed data processing by moving computation closer to the source, enabling real-time analysis and decision-making. Edge devices are decentralized, which creates privacy and confidentiality concerns, especially when applying machine learning algorithms to sensitive data. Privacy-preserving machine learning methods for edge computing are examined in this research. Federated learning, homomorphic encryption, differential privacy, and secure aggregation are examined as data protection methods for network edge machine learning. A thorough study of these methods shows the challenges of balancing privacy, computational economy, and model correctness. Federated learning has promise for collaborative model training without raw data sharing, but communication overhead and convergence speed remain. A fictional healthcare use case shows how federated learning may be used to train collaborative models across many edge devices while protecting patient data. The case study stresses the necessity for sophisticated optimizations to overcome edge device limits and regulatory compliance. Federated learning algorithms, privacy-preserving procedures, and ethics must be improved, according to the research. Future directions include improving heterogeneous edge algorithms, addressing data ownership and consent ethics, and increasing model decision-making openness. This paper presents essential insights on privacy-preserving machine learning in edge computing and advocates for robust techniques for different edge environments. The paper emphasizes the importance of technological advances, ethical frameworks, and regulatory compliance for secure and privacy-aware machine learning in decentralized edge computing.

Corresponding Author:

Deni Kurniawan,
Program Studi Sistem Informasi,
Universitas Bina Sarana Infotmatika, Jakarta, DKI Jakarta,
Jl. Kramat Raya No.98, RT.2/RW.9, Kwitang, Daerah Khusus Ibukota Jakarta 10450, Indonesia,
Email: deni@bsi.ac.id

1. Introduction

In recent years, the proliferation of edge computing has revolutionized the way data is processed, analyzed, and utilized[1][2]. Edge computing represents a paradigm shift from traditional centralized cloud computing by bringing computation closer to the data source, typically at the network edge, encompassing IoT devices, sensors, and mobile devices[3][4]. This architectural shift offers reduced

latency, improved response times, and enhanced bandwidth utilization, making it particularly advantageous for applications requiring real-time processing and decision-making[5].

However, this distributed nature of computing at the edge raises critical concerns regarding the privacy and security of the sensitive data being processed[6][7][8]. Edge devices, often constrained by limited computational resources and communication bandwidth, handle a multitude of data types, including personal, healthcare, industrial, and proprietary information[9][10]. The utilization of machine learning algorithms on such sensitive data at the edge necessitates stringent measures to safeguard individual privacy and prevent unauthorized access or disclosure[11][6][1].

Traditional methods of data transmission and centralized processing pose inherent risks, as data traversing networks or residing in centralized repositories are susceptible to interception, breaches, or exploitation[12]. Moreover, the application of machine learning models directly on raw, unencrypted data at the edge raises concerns about data leakage and privacy violations, especially in scenarios where compliance with regulatory frameworks (such as GDPR, HIPAA, etc.) is imperative[13][14].

To address these challenges, various privacy-preserving techniques have emerged as promising solutions[15][16][17][18][19]. Federated learning enables collaborative model training across distributed edge devices without sharing raw data, while homomorphic encryption allows computation on encrypted data without revealing the underlying information[20][21]. Differential privacy introduces controlled noise to protect individual data points, while secure multi-party computation ensures joint computations without disclosing individual inputs[22][23][24].

Despite these advancements, the effective integration and optimization of privacy-preserving machine learning techniques within edge computing environments remain a complex and evolving research domain[25][26]. The unique constraints of edge devices—limited processing power, energy, and storage—require tailored solutions that strike a balance between privacy preservation, computational efficiency, and model accuracy[27][28].

Therefore, this research endeavors to delve deeper into these challenges, exploring the limitations of existing privacy-preserving methodologies in edge computing contexts. By understanding the intricacies of edge environments, assessing the performance trade-offs, and innovating new techniques, this study aims to contribute novel insights and methodologies to enhance the privacy-preserving capabilities of machine learning in edge computing. Ultimately, the goal is to empower edge devices to perform sophisticated machine learning tasks while upholding data privacy and security standards.

2. Method And Materials

This section will introduce the concept of Machine Learning-based models as the basic framework in this research and the preparations made to build a new method Privacy-Preserving in Edge Computing Environments.

2.1. Machine Learning-based models

Machine Learning (ML) is a subset of artificial intelligence that focuses on developing algorithms and statistical models that enable computers to perform tasks without being explicitly programmed[29]. The core theory of machine learning encompasses various concepts, including supervised learning, unsupervised learning, and reinforcement learning[30][31][32].

a. Supervised Learning.

In supervised learning, the algorithm is trained on a labeled dataset, where each input is associated with a corresponding output label[33][34]. The goal is to learn a mapping from inputs to outputs so that the algorithm can make predictions on new, unseen data[33]. A common objective is to minimize the difference between predicted outputs and true labels, often measured using a loss function[35].

Basic Formula:

$$\boldsymbol{Loss} = \frac{\mathbf{1}}{\boldsymbol{n}} \sum_{\boldsymbol{i=1}}^{\boldsymbol{n}} \boldsymbol{\mathcal{L}}\ (\boldsymbol{y_i}, \boldsymbol{\hat{y}_i}) \quad \text{..... (1)}$$

Where $\boldsymbol{n}$ is the number of data points, $\boldsymbol{y_i}$ is the true label, $\boldsymbol{\hat{y}_i}$ is the predicted output, and $\boldsymbol{\mathcal{L}}$ is the loss function.

b. Unsupervised Learning.

Unsupervised learning deals with unlabeled data, aiming to find patterns, structures, or relationships within the data[36][37]. Clustering and dimensionality reduction are common tasks in unsupervised learning.

Basic Formula (K-Means Clustering):

$$\boldsymbol{\mathcal{J}} = \sum_{\boldsymbol{i=1}}^{\boldsymbol{k}} \sum_{\boldsymbol{j=1}}^{\boldsymbol{n}} \| \boldsymbol{x_j} - \boldsymbol{\mu_i} \|^2 \quad \text{..... (2)}$$

Where $\boldsymbol{k}$ is the number of clusters, $\boldsymbol{n}$ is the number of data points, $\boldsymbol{x_j}$ is a data point, $\boldsymbol{\mu_i}$ is the centroid of cluster $\boldsymbol{i}$, and ||·|| denotes the Euclidean distance.

c. Reinforcement Learning.

Reinforcement learning involves an agent learning to make decisions by interacting with an environment[38][39]. The agent receives feedback in the form of rewards or penalties based on its actions, allowing it to learn optimal strategies.

Basic Formula (Q-Learning):

$$\boldsymbol{Q(s,a)} = (\mathbf{1} - \boldsymbol{\alpha}) \cdot \boldsymbol{Q(s,a)} + \boldsymbol{\alpha} \cdot \left(\boldsymbol{R} + \boldsymbol{\gamma} \cdot \boldsymbol{max_{a'}}\, \boldsymbol{Q(s', a')}\right) \quad \text{..... (3)}$$

Where $\boldsymbol{Q(s,a)}$ is the quality of action $\boldsymbol{a}$ in state $\boldsymbol{s}$, $\boldsymbol{\alpha}$ is the learning rate, $\boldsymbol{R}$ is the immediate reward, $\boldsymbol{\gamma}$ is the discount factor, $\boldsymbol{s'}$ is the next state, and $\boldsymbol{a'}$ is the next action.

2.2. Developed mathematical model for privacy-preserving machine learning new mathematical model for privacy-preserving machine learning.

A mathematical model for privacy-preserving machine learning in edge computing involves defining the key components, variables, constraints, and objectives related to preserving privacy while performing machine learning tasks at the edge. Below is an abstract representation of a mathematical formulation for a privacy-preserving machine learning scenario in an edge computing environment.

Let's consider the scenario where a machine learning model is trained using data from multiple edge devices without sharing raw data but only model updates. The objective is to optimize the model training process while ensuring privacy through federated learning.

Symbols and Notations

$\boldsymbol{N}$ =Number of edge devices participating in the federated learning process.
$\boldsymbol{M}$ = Number of model parameters.
$\boldsymbol{D_i}$ = Dataset available at edge device $\boldsymbol{i}$.
$\boldsymbol{\theta}$ = Model parameters to be learned.
$\boldsymbol{U_i}$ = Local update computed by edge device $\boldsymbol{i}$ based on its dataset $\boldsymbol{D_i}$.
$\boldsymbol{w}$ = Aggregated model update received by the central server.

Variables:

$\boldsymbol{\theta_i}$ = Local model parameters at edge device $\boldsymbol{i}$.
$\boldsymbol{\alpha_i}$ = Weighting factor for edge device $\boldsymbol{i}$ during model aggregation.

Mathematical Formulation:

The federated learning process aims to minimize the discrepancy between the global model parameters $\boldsymbol{\theta}$ and the locally computed model updates across all edge devices while considering the privacy constraints.

The objective function can be formulated as a minimization problem:

$$\boldsymbol{min_{\theta}} \sum_{i=1}^{N} \boldsymbol{\alpha_i} \cdot \| \boldsymbol{\theta} - \boldsymbol{\theta_i} \|_2^2 \quad \text{..... (4)}$$

subject to:

$$w = \sum_{i=1}^{N} \alpha_i \cdot U_i$$

$$U_i = LocalUpdate\ (\theta_i, D_i)$$

Privacy Constraints: (e.g., differential privacy, encryption, noise addition).

The objective function aims to minimize the squared Euclidean distance between the global model parameters ($\boldsymbol{\theta}$) and the local model parameters (θ_i) weighted by α_i representing the importance of each edge device's contribution. The model updates (U_i) are computed locally at each device based on its dataset D_i.

The central server aggregates the model updates (w) received from the edge devices, considering privacy-preserving mechanisms such as differential privacy or encryption to ensure the protection of sensitive data during the aggregation process.

The exact formulation and constraints might vary depending on the specific privacy-preserving techniques employed (e.g., homomorphic encryption, differential privacy, secure aggregation) and the nature of the machine learning algorithm used.

2.3. Numerical example

A simplified numerical example illustrating federated learning for privacy-preserving machine learning in an edge computing scenario involving three edge devices.

Consider a scenario where a global model θ with two parameters ($M = 2$) is being trained across three edge devices ($N = 3$). Each edge device has its own dataset (D_1, D_2, D_3), and the objective is to update the global model through federated learning while preserving privacy.

For simplicity, let's assume the initial global model parameters are $\theta = [1,1]$, and the local model parameters at each edge device are as follows:

Edge device 1 (θ_1): $\theta_1 = [1.2,1.1]$
Edge device 2 (θ_2): $\theta_2 = [0.9,1.3]$
Edge device 3 (θ_3): $\theta_3 = [1.0,1.0]$

Let's use a simple averaging approach $\alpha_i = \frac{1}{N}$ for the weights to aggregate the model updates. The model update (U_i) for each device is the difference between its local parameters and the global parameters:

$$U_i = \theta - \theta_i$$

$$U_1 = [1.1] - [1.2,1.1] = [-0.2, -0.1]$$
$$U_2 = [1.1] - [0.9,1.3] = [0.1, -0.3]$$
$$U_3 = [1.1] - [1.0,1.0] = [0,0$$

Now, considering the objective function:

$$\boldsymbol{min_{\theta}} \sum_{i=1}^{N} \boldsymbol{\alpha_i} \cdot \| \boldsymbol{\theta} - \boldsymbol{\theta_i} \|_2^2$$

Let's update the global model parameters (θ) by aggregating the model updates (U_i) received from the edge devices:

$$w = \frac{1}{3} \cdot (U_1 + U_2 + U_3)$$
$$w = \frac{1}{3} \cdot ([-0.2, -0.1] + [0.1, -0.3] + [0.0])$$
$$w = \frac{1}{3} \cdot [0.1, -0.4]$$
$$w = [-0.033, -0.133]$$

Now, let's update the global model parameters:
$\theta_{new} = \theta + w$
$\theta_{new} = [1,1] + [-0.033, -0.133]$
$\theta_{new} = [0.967, -0.867]$

These updated parameters (θ_{new}) represent the new global model after aggregating the model updates from the edge devices while preserving privacy through federated learning.

This simplified example demonstrates the iterative process of federated learning where local model updates from edge devices are aggregated to update the global model while preserving the privacy of individual datasets at the edge.

3. Results And Discussion

A practical case where the mathematical model formulation for federated learning in edge computing, as described earlier, is applied to a scenario involving healthcare data.

Scenario: Federated Learning for Health Monitoring

In this hypothetical scenario, a healthcare provider aims to develop a machine learning model for predicting patient health risks using data collected from wearable health monitoring devices. The goal is to create a predictive model while ensuring the privacy of individual patient data stored on different edge devices.

Edge Devices: Three hospitals or medical centers each equipped with wearable health monitoring devices.
Data: Each hospital has a dataset (D_1, D_2, D_3) containing anonymized patient health data (e.g., vital signs, activity levels, medical history).
Objective: Develop a predictive model (θ) for identifying potential health risks without sharing raw patient data among the hospitals.

Numerical Example Recap:

Let's reuse the numerical example where three edge devices ($N = 3$) contribute their local model updates (U_1, U_2, U_3) to update the global model (θ) using federated learning.

Local model parameters:
Edge device 1 (θ_1): $\theta_1 = [1.2,1.1]$
Edge device 2 (θ_2): $\theta_2 = [0.9,1.3]$
Edge device 3 (θ_3): $\theta_3 = [1.0,1.0]$

Case Study Application:

1) Data Collection and Privacy Preservation: Each hospital utilizes federated learning, allowing the local models ($\theta_1, \theta_2, \theta_3$) to compute their updates based on their respective patient datasets (D_1, D_2, D_3).
2) Model Update and Aggregation: The local model updates (U_1, U_2, U_3) are sent to a centralized server for aggregation while preserving individual data privacy using secure aggregation techniques.
3) Global Model Update: The global model (θ) is updated by aggregating the local model updates according to the federated learning formulation.
4) Result: After model aggregation, the updated global model parameters (θ_{new}) are obtained (e.g., $\theta_{new} = [0.967, 0.867]$).

Application Outcome:
The federated learning process allows the healthcare provider to update the global predictive model (θ) without explicitly sharing patient data between hospitals. The updated model (θ_{new}) can now be used

to make predictions for potential health risks among patients while maintaining the privacy of sensitive health information stored at the edge devices.

This application demonstrates how federated learning, as formulated mathematically, can be practically applied in a healthcare context, enabling collaborative model training across multiple edge devices while ensuring data privacy and confidentiality.

4. Conclusion

The study embarked on an exploration of privacy-preserving machine learning methodologies in the context of edge computing environments, aiming to address the critical need for safeguarding sensitive data while leveraging the potential of machine learning at the network edge. Through a comprehensive review of existing techniques such as federated learning, homomorphic encryption, differential privacy, and secure aggregation, this research highlighted the significance of preserving data privacy in scenarios where decentralized edge devices process sensitive information. The investigation unveiled the complexities involved in balancing computational efficiency, model accuracy, and privacy constraints in edge environments. While federated learning emerged as a promising approach for collaborative model training without sharing raw data, challenges related to communication overhead, convergence speed, and scalability were identified. Moreover, the integration of privacy-preserving mechanisms into machine learning models at the edge necessitated a nuanced understanding of algorithmic trade-offs and computational constraints specific to resource-constrained devices. The numerical example and hypothetical application of a healthcare use case illustrated the practical implementation of federated learning for predictive model training while safeguarding individual patient data stored across multiple edge devices. However, it's imperative to acknowledge that real-world applications would demand more intricate privacy-preserving techniques and optimization strategies tailored to specific domains and regulatory frameworks. The study underscores the critical importance of advancing research and development in privacy-preserving methodologies to foster trust, compliance, and adoption of machine learning in edge computing. Future avenues of exploration might delve deeper into optimizing federated learning algorithms, refining differential privacy mechanisms, and devising novel approaches to mitigate privacy risks while maintaining model accuracy in diverse edge environments. In conclusion, the research serves as a foundational stepping stone toward establishing robust, privacy-aware machine learning frameworks tailored for edge computing, fostering a future where data confidentiality and computational advancements converge seamlessly at the network edge.

References

[1] D. Liu, Z. Yan, W. Ding, and M. Atiquzzaman, "A survey on secure data analytics in edge computing," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4946–4967, 2019.

[2] J.-H. Huh and Y.-S. Seo, "Understanding edge computing: Engineering evolution with artificial intelligence," *IEEE Access*, vol. 7, pp. 164229–164245, 2019.

[3] W. Rafique, L. Qi, I. Yaqoob, M. Imran, R. U. Rasool, and W. Dou, "Complementing IoT services through software defined networking and edge computing: A comprehensive survey," *IEEE Commun. Surv. Tutorials*, vol. 22, no. 3, pp. 1761–1804, 2020.

[4] N. A. Angel, D. Ravindran, P. M. D. R. Vincent, K. Srinivasan, and Y.-C. Hu, "Recent advances in evolving computing paradigms: Cloud, edge, and fog technologies," *Sensors*, vol. 22, no. 1, p. 196, 2021.

[5] S. Shukla, M. F. Hassan, L. T. Jung, and A. Awang, "Architecture for latency reduction in healthcare internet-of-things using reinforcement learning and fuzzy based fog computing," in *Recent Trends in Data Science and Soft Computing: Proceedings of the 3rd International Conference of Reliable Information and Communication Technology (IRICT 2018)*, Springer, 2019, pp. 372–383.

[6] A. Alwarafy, K. A. Al-Thelaya, M. Abdallah, J. Schneider, and M. Hamdi, "A survey on security and privacy issues in edge-computing-assisted internet of things," *IEEE Internet Things J.*, vol. 8, no. 6, pp. 4004–4022, 2020.

[7] J. Zhang, B. Chen, Y. Zhao, X. Cheng, and F. Hu, "Data security and privacy-preserving in edge computing paradigm: Survey and open issues," *IEEE access*, vol. 6, pp. 18209–18237, 2018.

[8] B. Ali, M. A. Gregory, and S. Li, "Multi-access edge computing architecture, data security and privacy: A

review," *IEEE Access*, vol. 9, pp. 18706–18721, 2021.
[9] V. Hayyolalam, M. Aloqaily, Ö. Özkasap, and M. Guizani, "Edge-assisted solutions for IoT-based connected healthcare systems: A literature review," *IEEE Internet Things J.*, vol. 9, no. 12, pp. 9419–9443, 2021.
[10] K. Toczé and S. Nadjm-Tehrani, "A taxonomy for management and optimization of multiple resources in edge computing," *Wirel. Commun. Mob. Comput.*, vol. 2018, 2018.
[11] U. A. Butt *et al.*, "A review of machine learning algorithms for cloud computing security," *Electronics*, vol. 9, no. 9, p. 1379, 2020.
[12] H. Tabrizchi and M. Kuchaki Rafsanjani, "A survey on security challenges in cloud computing: issues, threats, and solutions," *J. Supercomput.*, vol. 76, no. 12, pp. 9493–9532, 2020.
[13] J. Geng, "Taking Computation to Data: Integrating Privacy-preserving AI techniques and Blockchain Allowing Secure Analysis of Sensitive Data on Premise," 2023.
[14] J. Grover and R. Misra, "Keeping it Low-Key: Modern-Day Approaches to Privacy-Preserving Machine Learning," in *Data Protection in a Post-Pandemic Society: Laws, Regulations, Best Practices and Recent Solutions*, Springer, 2023, pp. 49–78.
[15] H. Ren, H. Li, Y. Dai, K. Yang, and X. Lin, "Querying in internet of things with privacy preserving: Challenges, solutions and opportunities," *IEEE Netw.*, vol. 32, no. 6, pp. 144–151, 2018.
[16] J. B. Bernabe, J. L. Canovas, J. L. Hernandez-Ramos, R. T. Moreno, and A. Skarmeta, "Privacy-preserving solutions for blockchain: Review and challenges," *IEEE Access*, vol. 7, pp. 164908–164940, 2019.
[17] A. Boulemtafes, A. Derhab, and Y. Challal, "A review of privacy-preserving techniques for deep learning," *Neurocomputing*, vol. 384, pp. 21–45, 2020.
[18] R. Xu, N. Baracaldo, and J. Joshi, "Privacy-preserving machine learning: Methods, challenges and directions," *arXiv Prepr. arXiv2108.04417*, 2021.
[19] S. Chenthara, K. Ahmed, H. Wang, and F. Whittaker, "Security and privacy-preserving challenges of e-health solutions in cloud computing," *IEEE access*, vol. 7, pp. 74361–74382, 2019.
[20] H. G. Abreha, M. Hayajneh, and M. A. Serhani, "Federated learning in edge computing: a systematic survey," *Sensors*, vol. 22, no. 2, p. 450, 2022.
[21] B. Zhang, G. Lu, P. Qiu, X. Gui, and Y. Shi, "Advancing Federated Learning through Verifiable Computations and Homomorphic Encryption," *Entropy*, vol. 25, no. 11, p. 1550, 2023.
[22] J. Böhler, "Input Secrecy & Output Privacy: Efficient Secure Computation of Differential Privacy Mechanisms." Karlsruhe Institute of Technology, Germany, 2021.
[23] K. K. K. Thissen, I. L. Schoenmakers, I. R. Koster, and I. P. van Liesdonk, "Achieving differential privacy in secure multiparty computation." Master's Thesis, Technische Universiteit Eindhoven, Eindhoven, 2019.
[24] A.-T. Tran, T.-D. Luong, J. Karnjana, and V.-N. Huynh, "An efficient approach for privacy preserving decentralized deep learning models based on secure multi-party computation," *Neurocomputing*, vol. 422, pp. 245–262, 2021.
[25] K. Sundarakantham and E. Sivasankar, "A hybrid deep learning framework for privacy preservation in edge computing," *Comput. Secur.*, vol. 129, p. 103209, 2023.
[26] L. Gao, T. H. Luan, B. Gu, Y. Qu, and Y. Xiang, *Privacy-preserving in edge computing*. Springer, 2021.
[27] S. Deng, H. Zhao, W. Fang, J. Yin, S. Dustdar, and A. Y. Zomaya, "Edge intelligence: The confluence of edge computing and artificial intelligence," *IEEE Internet Things J.*, vol. 7, no. 8, pp. 7457–7469, 2020.
[28] C. Jiang *et al.*, "Energy aware edge computing: A survey," *Comput. Commun.*, vol. 151, pp. 556–580, 2020.
[29] A. K. Tyagi and P. Chahal, "Artificial intelligence and machine learning algorithms," in *Research Anthology on Machine Learning Techniques, Methods, and Applications*, IGI Global, 2022, pp. 421–446.
[30] J. Alzubi, A. Nayyar, and A. Kumar, "Machine learning from theory to algorithms: an overview," in *Journal of physics: conference series*, IOP Publishing, 2018, p. 12012.
[31] E. F. Morales and H. J. Escalante, "A brief introduction to supervised, unsupervised, and reinforcement learning," in *Biosignal processing and classification using computational learning and intelligence*, Elsevier, 2022, pp. 111–129.
[32] M. Alloghani, D. Al-Jumeily, J. Mustafina, A. Hussain, and A. J. Aljaaf, "A systematic review on supervised and unsupervised machine learning algorithms for data science," *Supervised unsupervised Learn. data Sci.*, pp. 3–21, 2020.
[33] P. C. Sen, M. Hajra, and M. Ghosh, "Supervised classification algorithms in machine learning: A survey and review," in *Emerging Technology in Modelling and Graphics: Proceedings of IEM Graph 2018*, Springer, 2020, pp. 99–111.
[34] A. Iscen, G. Tolias, Y. Avrithis, and O. Chum, "Label propagation for deep semi-supervised learning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 5070–5079.

[35] Y. Ho and S. Wookey, "The real-world-weight cross-entropy loss function: Modeling the costs of mislabeling," *IEEE access*, vol. 8, pp. 4806–4813, 2019.
[36] B. Johnston, A. Jones, and C. Kruger, *Applied Unsupervised Learning with Python: Discover hidden patterns and relationships in unstructured data with Python*. Packt Publishing Ltd, 2019.
[37] A. A. Patel, *Hands-on unsupervised learning using Python: how to build applied machine learning solutions from unlabeled data*. O'Reilly Media, 2019.
[38] K. Zhang, Z. Yang, and T. Başar, "Multi-agent reinforcement learning: A selective overview of theories and algorithms," *Handb. Reinf. Learn. Control*, pp. 321–384, 2021.
[39] S. Gronauer and K. Diepold, "Multi-agent deep reinforcement learning: a survey," *Artif. Intell. Rev.*, pp. 1–49, 2022.