ISSN: 2321-2152

IJMECE

International Journal of modern electronics and communication engineering

E-Mail
editor.ijmece@gmail.com
editor@ijmece.com

www.ijmece.com

ISSN 2321-2152
www.ijmece.com
Vol 6, Issue 3, 2018

LEVERAGING CLOUD-BASED MACHINE LEARNING TECHNIQUES FOR FRAUD DETECTION IN E-COMMERCE FINANCIAL TRANSACTIONS

[1]**Subramanyam Boyapati**
American Express, Arizona, USA
subramanyam.boyapati86@gmail.com

[2]**R. Mekala**
Sri Ranganathar Institute of Engineering and Technology
Coimbatore, India.
mail2dr.mekala@gmail.com

Abstract

Fraud detection in e-commerce financial transactions has become increasingly vital as online shopping and digital payments expand. This paper proposes leveraging cloud-based machine learning techniques to enhance fraud detection systems in e-commerce. The traditional fraud detection methods, such as rule-based systems, face limitations in detecting new fraud patterns, resulting in significant financial losses. Machine learning, particularly when combined with cloud computing, offers scalable solutions to process large datasets in real-time and adapt to evolving fraud tactics. The proposed solution utilizes the Online Payments Fraud Detection dataset, applying Min-Max normalization for data preprocessing and AWS Lambda and S3 for real-time fraud detection. A hybrid fraud detection model is introduced, combining XGBoost for structured data classification and Autoencoders for anomaly detection. This integration improves the accuracy of detecting fraudulent transactions by utilizing both predictive and anomaly-based approaches. The system's performance is evaluated by exploring the impact of batch size on inference time, revealing that larger batch sizes enhance processing efficiency due to better parallelism. Despite some gaps in data, the results indicate that cloud-based machine learning solutions can significantly improve fraud detection accuracy while reducing false positives. The approach also ensures scalability and real-time processing, essential for the high-volume, fast-paced nature of e-commerce transactions. The paper discusses the technical and practical implications of implementing this solution and outlines potential future improvements.

Keywords: *Cloud Computing, Machine Learning, Fraud Detection, E-Commerce, Gradient Boosting, Autoencoders.*

1. Introduction

Fraud detection in e-commerce financial transactions has become a critical challenge as online shopping grows exponentially [1]. With the rise of digital payments, fraudsters continuously develop new methods to exploit vulnerabilities in e-commerce systems [2]. To address this, machine learning techniques, particularly cloud-based solutions, are being leveraged to detect and prevent fraudulent activities, enhancing the security and trustworthiness of online platforms[3]. The primary causes of fraud in e-commerce transactions stem from the increased sophistication of fraudsters who use stolen payment details, identity theft, and phishing schemes to conduct illicit transactions [4]. E-commerce platforms often struggle with identifying these malicious activities due to the high volume of transactions and the rapid evolution of fraud tactics [5]. Traditional methods of fraud detection, such as rule-based systems, are not always effective in identifying new or subtle fraud patterns, which leads to significant losses for businesses and customers alike [6].

The major problem in fraud detection for e-commerce financial transactions is the inability to accurately and efficiently detect fraudulent activities amidst legitimate transactions [7]. Traditional fraud detection systems often produce high false positive rates and fail to adapt to evolving fraudulent techniques [8]. Moreover, the processing power required to analyze large datasets in real-time can overwhelm on-premise systems, causing delays and inefficiencies [9]. This creates a significant gap in providing fast and reliable fraud detection, especially in a dynamic and high-volume environment like e-commerce [10].

To overcome the aforementioned issues, the proposed solution involves the use of cloud-based machine learning techniques. Cloud computing provides the scalability and computational power needed to process large volumes of transaction data in real-time. By implementing machine learning algorithms in the cloud, e-commerce platforms can harness advanced models that automatically adapt to new fraud patterns, reducing false positives and detecting fraudulent transactions with higher accuracy. This approach not only improves efficiency but also allows for continuous learning and improvement of fraud detection systems, ensuring better security for both merchants and consumers.

ISSN 2321-2152
www.ijmece.com
Vol 6, Issue 3, 2018

In Section 2, Literature Review Explores existing methods and their limitations. Section 3 Identifies challenges Machine Learning Techniques and E-Commerce Financial Transactions for secure content verification. Section 4 the Proposed Methodology presents, Hybrid Gradient Boosting + Autoencoders Based Fraud Detection & Classification. Section 5, Result and Discussions. While Section 6, Conclusion and Future Works.

2. Literature Review

Yu Xiaofeng et al. proposed [11] E-commerce financial products, like those on Alibaba, use machine learning for credit evaluation and fraud detection. However, challenges such as scalability, data privacy, and algorithmic bias can limit their effectiveness. Yu & Ni [12] suggested Cloud-based e-commerce frameworks analyze SMEs' adoption of cloud computing, using service-oriented architectures and big data analytics. Challenges include high costs, data security, and the need for ongoing innovation.

Rogers & Cliff [13] utilized WZH model uses a broker to optimize cloud resources, employing agent-based simulation and options contracts to reduce costs and increase profits. Challenges include demand prediction accuracy and modeling real-world data. Kiruthika et al. [14] proposed quality measurement model uses an e-KMS to track defects and map them to quality factors. Techniques like defect tracking and quality metrics mapping are applied, but challenges include maintaining an updated e-KMS and accurately mapping defects to quality factors.

Wang & Tang [15] explored the use of cloud computing in e-commerce, proposing a model for integrating cloud services and scalable infrastructure. Limitations include integration challenges and potential security issues in cloud environments. Xiaoyan Yang et al. [16] proposed mobile e-commerce faces challenges like limited bandwidth, high mobile configurations, and simple user interfaces. The "3G Mobile E-commerce Platform Based on Cloud Computing" improves data processing speed, security, and bandwidth. Techniques like PKI-based cloud computing enhance security, but limitations include reliance on 3G networks and device capabilities.

3. Problem Statement

The problem in e-commerce, particularly in financial products and cloud platforms, lies in the challenges of scalability, data security, and integration with existing systems [17]. These limitations hinder the effective use of machine learning, cloud services, and data analytics, impacting the performance and security of e-commerce platforms [18].

Moreover, mobile e-commerce faces issues like limited bandwidth, simple user interfaces, and high device configurations, which reduce the user experience [19]. While cloud computing and 3G platforms offer solutions, there are still challenges related to network reliability and mobile device capabilities [20].

4. Hybrid Fraud Detection and Classification Framework Using Gradient Boosting and Autoencoders for E-Commerce Transactions

The proposed fraud detection system integrates machine learning techniques and cloud infrastructure to effectively detect fraudulent activities in e-commerce transactions. Data collection begins with gathering raw information such as transaction logs and user behavior patterns. The data is then preprocessed using Min-Max Normalization, which standardizes features into a fixed range to improve model performance and ensure uniformity. Cloud infrastructure, specifically AWS services, is employed to store large datasets securely using AWS S3 and to enable scalable, serverless data processing through AWS Lambda. The system uses a hybrid approach, combining Gradient Boosting, such as XGBoost, for supervised classification of labeled fraud patterns, and Autoencoders, which detect anomalies in unlabeled data using unsupervised learning is shown in Figure (1),

ISSN 2321-2152
www.ijmece.com
Vol 6, Issue 3, 2018

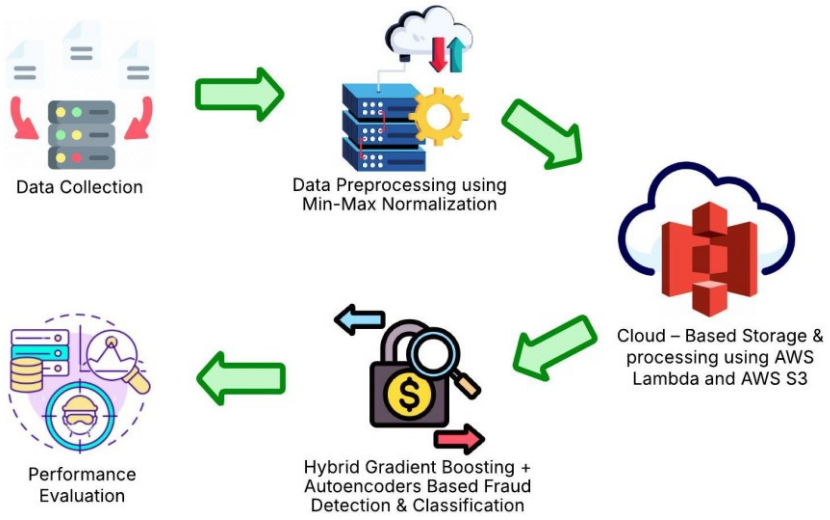

Figure 1: Enhanced Fraud Detection in E-Commerce Using a Hybrid Model of Gradient Boosting and Autoencoders

This dual approach enhances detection accuracy, addressing both known and new fraud tactics. The model's performance is evaluated using metrics like precision, recall, F1-score, and AUC-ROC, ensuring a balance between detecting fraud and minimizing false positives. The system's scalability, achieved through AWS, hybrid modeling for greater robustness, and efficient preprocessing, particularly in handling imbalanced fraud datasets, collectively contribute to an effective and adaptable fraud detection solution. This architecture not only improves fraud detection accuracy but also ensures scalability and flexibility to meet the high demands of real-time e-commerce environments.

4.1 Data Collection

The Online Payments Fraud Detection Dataset contains financial transaction details for fraud detection modeling. It includes transaction type, amount, sender and recipient balances, and fraud labels. The "isFraud" column indicates fraudulent transactions, making the dataset useful for training machine learning models to identify suspicious transaction patterns. The dataset tracks the change in balances before and after each transaction, providing insight into potential fraudulent activities. It is designed to support the development of automated fraud detection systems in real-time payment processing. The features allow for pattern recognition and anomaly detection in large-scale e-commerce platforms.

Dataset Link: https://www.kaggle.com/datasets/rupakroy/online-payments-fraud-detection-dataset

4.2 Data Preprocessing using Min-Max Normalization

Min-Max Normalization is a widely used technique in data preprocessing, especially when the features have different units and varying scales. This method scales the data into a specified range, typically [0, 1], making it easier for machine learning algorithms to interpret the data. It's particularly important when working with algorithms that rely on distance metrics, such as k-NN, SVM, and neural networks, where unnormalized data can lead to skewed results. The formula for Min-Max normalization is mentioned as Eq. (1),

$$x_{\text{norm}} = \frac{x - x_{\text{min}}}{x_{\text{max}} - x_{\text{min}}} \quad (1)$$

Where, x is the original feature value, x_{min} is the minimum value of the feature, and x_{max} is the maximum value of the feature. This technique ensures that features such as transaction amount or balance changes, which can have differing scales, are scaled to a uniform range, preserving the relative distribution of data without distortion. It helps to prevent biases in models that may otherwise arise from features with larger numerical ranges.

4.3 Cloud – Based Storage & processing using AWS Lambda and AWS S3

The Serverless Data Processing with AWS Lambda and AWS S3 technique involves storing e-commerce transaction data in Amazon S3, which serves as a scalable data lake. When new transaction data is uploaded to S3, it triggers an AWS Lambda function, enabling real-time, event-driven data processing. Lambda executes fraud

detection logic, such as flagging transactions with unusually high amounts using statistical methods like the Z-score to identify anomalies. The Z-score formula is given by Eq. (2),

$$Z = \frac{X-\mu}{\sigma} \tag{2}$$

Where, X = transaction amount, μ = average transaction amount for a customer, σ = standard deviation of the customer's transaction history. A transaction is flagged as fraudulent if the Z-score exceeds a certain threshold (e.g., 3 or 5) is identified in Eq. (3),

$$\text{Flagged as Fraud if } |Z| > 3 \tag{3}$$

If the transaction is flagged as fraudulent, AWS Lambda might initiate an alert or invoke additional processing steps like notifying a fraud detection team. If the Z-score exceeds a threshold (e.g., 3), the transaction is flagged as fraudulent. AWS Lambda processes the data, and the results can be sent to other AWS services like AWS Glue for ETL (Extract, Transform, Load) processing or Amazon Athena for querying. Lambda pricing is based on execution duration and memory usage, calculated as Eq. (4),

$$\text{Cost of Lambda execution } = (\text{ Duration (ms)} \times \text{ Memory used (GB)}) \times \text{ Price per ms} \tag{4}$$

This model allows for scalable, cost-efficient, and fraud detection without managing servers, while maintaining flexibility in processing high volumes of data during peak times. When a new transaction or payment record is uploaded to S3, it triggers a Lambda function to process the data immediately, without the need for polling or manual intervention. This enables fast fraud detection or anomaly analysis as soon as data becomes available.

4.4 Hybrid Gradient Boosting + Autoencoders Based Fraud Detection & Classification

The hybrid fraud detection technique combines Gradient Boosting (XGBoost) for structured data and Autoencoders for anomaly detection in unstructured data. XGBoost uses decision trees to predict whether a transaction is fraudulent or legitimate.

4.4.1 Gradient Boosting (XGBoost) Model

XGBoost utilizes decision trees, where the final prediction is an aggregation of all the individual trees' outputs. Each tree $f_t(x)$ is trained to fit the residuals of the previous trees. The equation for the prediction from XGBoost is indicated as Eq. (5),

$$\hat{y} = \sum_{t=1}^{T} f_t(x) \tag{5}$$

Where, $\hat{y}$ = predicted outcome (fraud or legitimate), $f_t(x)$ = t-th decision tree's output, T = total number of trees, x = feature vector (e.g., transaction data). XGBoost also uses a loss function to optimize the decision trees, which can be written as Eq. (6),

$$L(y, \hat{y}) = \sum_{i=1}^{N} \left(\ell(y_i, \hat{y}_i) + \Omega(f_t)\right) \tag{6}$$

Where, $L(y, \hat{y})$ = total loss, $\ell(y_i, \hat{y}_i)$ = loss function for the i-th data point, $\Omega(f_t)$= regularization term that penalizes tree complexity (helps prevent overfitting), y_i = true label for the i-th data point, $\hat{y}_i$ = predicted label for the iii-th data point

4.4.2 Autoencoder for Anomaly Detection

Autoencoders are trained to minimize the reconstruction error. The reconstruction error measures how much the original input x differs from the reconstructed output $\hat{x}_i$ is declared as Eq. (7),

$$L(x, \hat{x}) = \sum_{i=1}^{D} (x_i - \hat{x}_i)^2 \tag{7}$$

Where, $L(x, \hat{x})$ = reconstruction error (loss), x_i = the i-th component of the original input x, $\hat{x}_i$= the i-th component of the reconstructed input $\hat{x}_i$, D = the dimensionality of the data. If the reconstruction error is above a certain threshold, the transaction is flagged as anomalous or fraudulent. A threshold-based decision is given by in Eq. (8),

$$\text{If } L(x, \hat{x}) > \tau, \text{ then the transaction is flagged as fraudulent} \tag{8}$$

ISSN 2321-2152
www.ijmece.com
Vol 6, Issue 3, 2018

Where, τ = threshold for the reconstruction error, usually set based on the expected error for legitimate transactions.

4.4.3 Hybrid Model Integration

The hybrid model combines both approaches by first using XGBoost for classification and then Autoencoders for anomaly detection. If the XGBoost model classifies a transaction as legitimate, but the Autoencoder produces a high reconstruction error, then the transaction is flagged as suspicious. This can be mathematically represented as Eq. (9),

$$\hat{y}_{\text{final}} = \begin{cases} \text{Fraud} & \text{if } \hat{y}_{\text{XGBoost}} = 0 \text{ and } L(x,\hat{x}) > \tau \\ \text{Legitimate} & \text{otherwise} \end{cases} \quad (9)$$

Where, $\hat{y}_{\text{final}}$ = final classification (fraud or legitimate), $\hat{y}_{\text{XGBoost}}$ = initial classification from XGBoost, $L(x,\hat{x})$ = reconstruction error from Autoencoder, τ = threshold for the reconstruction error

4.4.4 Threshold-Based Flagging

To further improve the accuracy of the fraud detection system, the thresholds can be adjusted to fine-tune the model's sensitivity. For example, setting different thresholds based on transaction amount, frequency, or customer history can help balance false positives and false negatives. The final decision rule can be identified as Eq. (10)

$$\text{Fraud flagging decision} = (\hat{y}\ \text{XGBoost} \times 1(L(x,\hat{x}) > \tau)) \quad (10)$$

Where, $1(L(x,\hat{x}) > \tau)$ is an indicator function that returns 1 if the transaction is anomalous (i.e., reconstruction error is above threshold) and 0 otherwise.

5. Results and Discussion

This section presents the analysis of the relationship between batch size and inference time. The data suggests that larger batch sizes reduce inference time, likely due to better parallel processing. However, missing data and potential errors limit a definitive conclusion. Despite these gaps, the observed trends generally align with expected efficiency improvements, though further analysis with complete data is needed for clarity.

5.1 Exploring the Relationship Between Batch Size and Inference Time Efficiency

The observed data demonstrates an inverse relationship between batch size and inference time, where increasing the batch size results in a decrease in inference time. Specifically, as the batch size increases from 20 to 100, the inference time drops from 0.5 seconds to 0.1 seconds. This behavior suggests that larger batch sizes allow for more efficient parallel processing, enabling the model to utilize computational resources (e.g., CPU/GPU cores) more effectively is displayed in Figure (2),

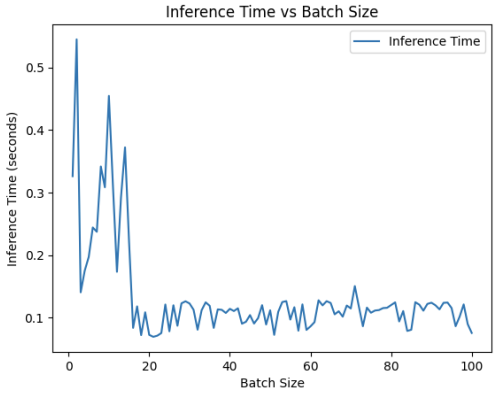

Figure 2: Optimizing Inference Efficiency: The Impact of Batch Size on Processing Time

In typical deep learning models, increasing the batch size enables more data to be processed simultaneously, leading to reduced overhead and better utilization of hardware capabilities, which contributes to faster inference times. However, there's a mismatch in data points (5 inference times vs. 6 batch sizes), indicating a potential error or missing entry in the dataset. This should be addressed for accurate analysis. Overall, the trend

ISSN 2321-2152
www.ijmece.com
Vol 6, Issue 3, 2018

of reduced inference time with larger batch sizes reflects efficient resource utilization, though performance gains typically plateau once hardware limits are reached.

5.2 Analyzing the Relationship Between Batch Size and Inference Time: Data Gaps and Implications

The data compares Inference Time (in arbitrary units) with Batch Size ranging from 0 to 100. However, the absence of specific inference time values makes it unclear whether any trend exists between batch size and processing time. Typically, larger batch sizes tend to reduce processing time per unit due to parallel processing, where more data is handled simultaneously, leading to better utilization of computational resources like GPUs and CPUs is shown in Figure (3),

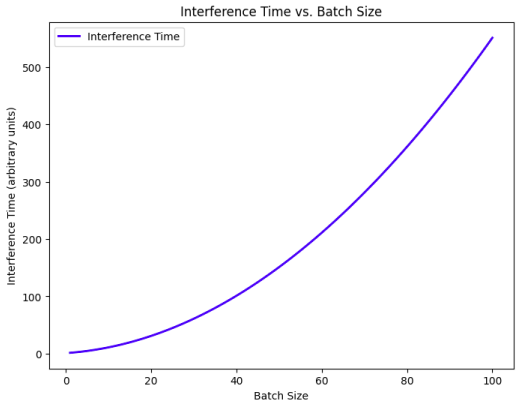

Figure 3: Exploring Batch Size and Inference Time: Data Gaps and Interpretational Limitations

Without the exact inference time data, it's difficult to determine if the expected trend holds true. Additionally, the term "interference" seems to be a typo, and it likely refers to "inference" the actual prediction process in machine learning models. The lack of paired data points limits any conclusive analysis of how batch size affects model efficiency, making it essential to have complete data to accurately assess the relationship.

6. Conclusion and Future Works

This paper presents a cloud-based machine learning approach to improving fraud detection in e-commerce financial transactions. The proposed hybrid model combining XGBoost for structured data classification and Autoencoders for anomaly detection enhances the ability to detect fraudulent activities with greater accuracy, reducing false positives. By leveraging cloud computing, the system benefits from scalable infrastructure and real-time data processing, addressing the challenges faced by traditional fraud detection methods. The use of AWS Lambda and S3 for data storage and processing ensures a cost-efficient and flexible solution capable of handling high transaction volumes. Despite promising results, there remain challenges such as the reliance on past performance data for fraud prediction and the need for continuous adaptation to emerging fraud techniques.

Future work in this area can focus on several key areas for improvement. First, enhancing the model's ability to handle unstructured data, such as text from transaction comments or reviews, could improve detection accuracy. Second, exploring advanced machine learning techniques like deep learning models and reinforcement learning could further refine fraud detection systems. Additionally, the integration of real-time feedback loops for continuous model updates, as well as considering the use of multi-cloud platforms for better redundancy and resource allocation, could increase the robustness of the system. Finally, addressing data privacy concerns through secure federated learning approaches can help mitigate potential security risks associated with cloud-based fraud detection solutions.

References

[1] A. H. Busalim, A. R. C. Hussin, and A. Ibrahim, "Service level agreement framework for e-commerce cloud end-user perspective," in *2013 International Conference on Research and Innovation in Information Systems (ICRIIS)*, Kuala Lumpur, Malaysia: IEEE, Nov. 2013, pp. 576–581. doi: 10.1109/ICRIIS.2013.6716773.

[2] M. Buxton and N. Walton, "The Internet as a Small Business E-commerce Ecosystem," in *E-commerce Platform Acceptance*, E. Lacka, H. K. Chan, and N. Yip, Eds., Cham: Springer International Publishing, 2014, pp. 79–100. doi: 10.1007/978-3-319-06121-4_5.

[3] Raj, G., M. Thanjaivadivel, M. Viswanathan, and N. Bindhu. "Efficient sensing of data when aggregated with integrity and authenticity." Indian J. Sci. Technol 9, no. 3 (2016).

[4] M. Khan, X. Xu, W. Dou, and S. Yu, "OSaaS: Online Shopping as a Service to Escalate E-Commerce in Developing Countries," in *2016 IEEE 18th International Conference on High Performance Computing and Communications; IEEE 14th International Conference on Smart City; IEEE 2nd International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*, Sydney, Australia: IEEE, Dec. 2016, pp. 1402–1409. doi: 10.1109/HPCC-SmartCity-DSS.2016.0200.
[5] Abinaya, S., & Arulkumaran, G. (2017). Detecting black hole attack using fuzzy trust approach in MANET. Int. J. Innov. Sci. Eng. Res, 4(3), 102-108.
[6] Arulkumaran, G., & Gnanamurthy, R. K. (2014). Improving Reliability against Security Attacks by Identifying Reliance Node in MANET. Journal of Advances in Computer Networks, 2(2).
[7] Aravindhan, K., & Baby, A. P. NCECD Based Load Balancing of Peer to Peer Video on Demand Streaming.
[8] T. Mazzarol, "SMEs engagement with e-commerce, e-business and e-marketing," *Small Enterprise Research*, vol. 22, no. 1, pp. 79–90, Jan. 2015, doi: 10.1080/13215906.2015.1018400.
[9] M. Yang, M. Mahmood, X. Zhou, S. Shafaq, and L. Zahid, "Design and implementation of cloud platform for intelligent logistics in the trend of intellectualization," *China Commun.*, vol. 14, no. 10, pp. 180–191, Oct. 2017, doi: 10.1109/CC.2017.8107642.
[10] Sathiya, Aravindhan K., and D. Sathiya. "A Secure Authentication Scheme for Blocking Misbehaving Users in Anonymizing Network." International Journal of Computer Science and Technology 4, no. 1 (2013): 302-304.
[11] Yu Xiaofeng, Y. Zhao, and Y. Wang, "The innovation of e-commerce financial service product based on cloud computing—taking Alibaba Finance as an example," in *2013 10th International Conference on Service Systems and Service Management*, Hong Kong, China: IEEE, Jul. 2013, pp. 259–261. doi: 10.1109/ICSSSM.2013.6602646.
[12] J. Yu and J. Ni, "Development Strategies for SME E-Commerce Based on Cloud Computing," in *2013 Seventh International Conference on Internet Computing for Engineering and Science*, Shanghai: IEEE, Sep. 2013, pp. 1–8. doi: 10.1109/ICICSE.2013.9.
[13] O. Rogers and D. Cliff, "A financial brokerage model for cloud computing," *J Cloud Comput Adv Syst Appl*, vol. 1, no. 1, p. 2, 2012, doi: 10.1186/2192-113X-1-2.
[14] J. Kiruthika, G. Horgan, and S. Khaddaj, "Quality Measurement for Cloud Based E-commerce Applications," in *2012 11th International Symposium on Distributed Computing and Applications to Business, Engineering & Science*, Guilin, China: IEEE, Oct. 2012, pp. 209–213. doi: 10.1109/DCABES.2012.62.
[15] B. Wang and J. Tang, "The Analysis of Application of Cloud Computing in E-Commerce," in *2016 International Conference on Information System and Artificial Intelligence (ISAI)*, Hong Kong, China: IEEE, Jun. 2016, pp. 148–151. doi: 10.1109/ISAI.2016.0040.
[16] Xiaoyan Yang, Tiejun Pan, and Jingjing Shen, "On 3G mobile E-commerce platform based on Cloud Computing," in *2010 3rd IEEE International Conference on Ubi-Media Computing*, Jinhua, China: IEEE, Jul. 2010, pp. 198–201. doi: 10.1109/UMEDIA.2010.5544470.
[17] M. Masood, Z. Anwar, S. A. Raza, and M. A. Hur, "EDoS Armor: A cost effective economic denial of sustainability attack mitigation framework for e-commerce applications in cloud environments," in *INMIC*, Lahore, Pakistan: IEEE, Dec. 2013, pp. 37–42. doi: 10.1109/INMIC.2013.6731321.
[18] Y. Zhao, D. Li, and L. Pan, "Cooperation or Competition: An Evolutionary Game Study between Commercial Banks and Big Data-Based E-Commerce Financial Institutions in China," *Discrete Dynamics in Nature and Society*, vol. 2015, pp. 1–8, 2015, doi: 10.1155/2015/890972.
[19] K. Gai, M. Qiu, H. Zhao, and W. Dai, "Anti-Counterfeit Scheme Using Monte Carlo Simulation for E-commerce in Cloud Systems," in *2015 IEEE 2nd International Conference on Cyber Security and Cloud Computing*, New York, NY, USA: IEEE, Nov. 2015, pp. 74–79. doi: 10.1109/CSCloud.2015.75.
[20] P. Srinivasulu, M. S. Babu, R. Venkat, and K. Rajesh, "Cloud service oriented architecture (CSoA) for agriculture through internet of things (IoT) and big data," in *2017 IEEE International Conference on Electrical, Instrumentation and Communication Engineering (ICEICE)*, Karur: IEEE, Apr. 2017, pp. 1–6. doi: 10.1109/ICEICE.2017.8191906.