*Grenze International Journal of Engineering and Technology, June Issue*

# Smart Detection of Ad Click Fraud via Machine Learning and Deep Learning Algorithms

E. Ramesh[1], Mallisetty Jyothi[2], Kataru Meghana[3], Kamalapuram kalyani[4] and Bobburi Lalitha gangadevi[5]
[1]Assistant Professor, Dept of CSE, Annamacharya University Boyanpalli, Rajampet
Email: eeramesh8@gmail.com
[2-5]Student, Dept of CSE, Annamacharya institute of technology and sciences Boyanpalli, Rajampet
Email: jyothimallisetty6@gmail.com, katarumeghana24@gmail.com, kalyanikamalapuram5@gmail.com, lalithabobburi2005@gmail.com

***Abstract*— Online advertising is now a critical aspect of digital marketing, as it helps organizations to target the audience effectively. Nonetheless, an expanding ecosystem has also contributed to the increase in ad click fraud, as automatic bots or ill-intentioned individuals produce artificial clicks to inflate engagement data and generate at least thousands of dollars in losses. This paper introduces a smart and scalable model of Ad Click Fraud Detection based on both Machine Learning (ML) and Deep Learning (DL) methods. The given model utilizes the Online Advertising Digital Marketing Dataset of the Kaggle that has been preprocessed (data cleaning, encoding, normalization, and Synthetic Minority Oversampling Technique (SMOTE)) in this case to deal with the issue of class imbalance. Random Forest, Gradient Boosting, LightGBM, and Artificial Neural Network (ANN) models were compared and analyzed, and Random Forest had the best classification accuracy and strength. Moreover, a Flask-based web interface can be used to detect fraud in real-time and can be used to verify the authenticity of the selected clicks instantly. The suggested solution is very precise, scalable, and deployable and offers a viable way to overcome the problem of fraudulent ad engagements and protect advertising investment.**

***Keywords*— Ad Click Fraud Detection, Machine Learning, Deep Learning, Random Forest, LightGBM, SMOTE, Artificial Neural Network, Flask, Digital Marketing Analytics.**

## I. INTRODUCTION

Online advertisement has proved to be among the best business promotion instruments of digital transformation era. The Google Ads, Facebook and programmatic ad exchanges platforms are delivering billions of ad impressions daily, and are establishing enormous revenue streams. It is clear, however, with the increasing digital ad ecosystem that the sophistication of click fraud has increased as well; it is a form of fraud that involves automated bots or individuals with incentives in order to simulate ad clicks, not because people are interested in them. It is an inflated activity that artificially inflates the clicks and may leave the advertisers with a huge sum of money to purchase a huge amount of fake traffic. The strategic impact is not isolated but also the financial one, wherein the misrepresentation of data of the engagement influences the marketing decision-making, and the establishment of the return-on-investment (ROI).

Ad click fraud can be of different types. Bot-based fraud involves the usage of wizards that mimics the actions of users, and the usage of human beings to make artificial clicks in large numbers through click farms.

*Grenze ID: 01.GIJET.12.2.47*

Another form of fraud is publisher fraud in which web owners deliberately seek to make clicks to make more money on advertisements. To counter such threats, there is need to develop the detection systems which will take into account the complex information concerning the behavior and train to detect the invisible signs of fraud. The traditional rule-based techniques that involve the blockage of IPs with high number of clicks are not capable of handling the modern adaptive attackers whose mechanisms of covering fraud patterns are IP rotations, VPNs and randomized behavior.

The specified project will propose an AI-based hybrid system combining the advantages of the Machine Learning (ML) models and Deep Learning (DL) models to recognize the fraud and detect it in real-time and correctly. The dataset that will be utilized is Online Advertising Digital Marketing Data that contains detailed description, such as, campaign cost, number of displays, number of click, revenue and post- click conversions. These are necessary indicators of campaign sincerity and the system will distinguish between true conversions and dubious behaviors on the clicks.

The preprocessing stage involves various major steps. On the one hand, the duplication of data is avoided through the deletion of irrelevant or missing attributes. Second, the numeric encoder of Month, campaign number and position of banner is numeric and is based on the Label Encoding. Under Standard Scaling, numerical variables are scaled in order to equalize differences between the ranges in order to maximize the convergence of a model. Since the data in ad fraud is skewed (i.e. it consists of more normal clicks than of fraudulent clicks), Synthetic Minority Over-sampling Technique (SMOTE) is applied to generate more balanced training data to ensure that the model will not only learn successfully in the fraudulent classes, but also in normal classes.

A number of algorithms are forecasted and experimented. A random Forest, Gradient boosting and XGBoost are also applicable in the ML sector due to their ability to form an ensemble and the ability to capture any non-linear relationship. The best of them was random forest which had near-perfect accuracy, precision, and recall in the presence of SMOTE balancing. At the same time, more profound non-linear representations were checked with the help of Artificial Neural Networks (ANNs). However, ANN had an accuracy of only approximately 86 percent and the training time and interpretability was worse as compared to the collective ML methods, and thus Random Forest is more suitable to apply in real-time.

Flask based web-interface was developed to use the trained random forest model. The backend takes provided user data - displays, cost, revenue and post-click conversion, sales amount and generates an on-the-fly fraud prediction with certain preprocessing transformations (scaling, reshaping). The frontend provides a friendly interface that is easy to use where the advertisers are able to test various scenarios of the campaigns. The result of the prediction is in visual form of Genuine Click or Fraudulent Click, so as to be easily understood by the non-technical user.

Technically, the system demonstrates how the interaction of the effectiveness of ML and the flexibility of DL can be effective in terms of a single architecture. The accuracy, speed and usability trifecta is achieved in this project with built in data scaling, scaling ensemble modeling and web deployment. In a modular approach, the architecture has been developed so that it can accommodate future models without necessarily having to re-engineer it. Besides, with joblib serialization and Flask, the entire solution can be launched and run on a local machine or on cloud providers (AWS, Heroku) with extremely low latency.

To sum up, the given project can be regarded as a feasible application of Artificial Intelligence to digital marketing analytics. It enables advertisers to filter out fraudulent activities beforehand and safeguard their marketing budgets by relating data science to business requirements and enhance the legitimacy of campaign performance analytics.

## II. Literature Review

Detection of fraud in digital advertising has gone through a radical development during the last 10 years. The initial studies were mainly conducted on the basis of rule-based and statistical models which utilized user behavior and clickstream data to detect abnormal interaction patterns. Haddabi et al. [1] were one of the first to come up with a framework that tested the interval of clicks, repetition of IPs and the duration of sessions to make a distinction between legitimate users and automated bots. This method was useful in detecting simple anomalies but relied on fixed statistical values and hence was not flexible to detect new and complex fraud schemes. On the same note, Metwally et al. [2] suggested a real-time detection system that operates with behavioral fingerprinting and IP tracker. Their research implemented the adaptive filtering and temporal fingerprinting to connect the distributed fraud activities and made it possible to detect frauds in advertising networks very quickly. But privacy issues and the problems with cross-device correlation decreased its practicality in the long term.

The transition to machine learning made the fraud detection field data-driven and able to construct the complex non-linear relationships. By integrating classifiers, including Decision Trees and Logistic Regression, with both temporal and behavioral features, Zhou et al. [3] were the first to use supervised learning methods to detect online click frauds, and better precision and interpretability were obtained in comparison with traditional ones. It is based on this that Hamed and Aziz [4] proposed a hybrid ensemble framework that used a Support Vector machine (SVM) coupled with the Random Forest algorithm to improve detection efficiency as well as resilience to noisy and imbalanced data. Their retraining approach was a significant move towards lifelong learning systems that could manage the changing fraud trends of limited clicks.

In association with the development of deep learning, scientists started to utilize neural structures to not only capture high-dimensional dependencies in ad-clicks, but also capture sequential dependencies. Cao et al. [5] established that the Deep Neural Networks (DNNs) were capable of automatically learning latent features of user-advertisement interactions and performing better than classical models based on recall and F1-score. Yin et al. [6] built on the concept by modeling the sequences of clicks in two-dimensional matrices and using Convolutional Neural Networks (CNNs) to detect spatial-temporal patterns of user visits. Their architecture was successful in separating human clicking and organized bot interactions and focusing on the importance of deep architectures in acquiring implicit cues of fraud. However, the interpretability and computational complexity of deep networks made it difficult to be applied to industry at the time.

On the whole, contemporary literature proves the vivid line of research out of the models of a static set of rules to the adaptive and AI-powered models that can make real-time decisions. Although ensemble and deep learning models have been very accurate and scaled, they have problems in the context of continuous retraining, model explanability, and dealing with fraudulent behaviors that change very fast. These loopholes encourage the creation of a hybrid detection system based on combination of machine and deep learning algorithms, dataset balancing with SMOTE, and real-time detection with the help of the Flask-based interface-a method implemented in the offered system of the present analysis.

## III. Proposed Model

The proposed system will focus on creating an intelligent, scalable, and real-time Ad Click Fraud Detection Framework with the help of a hybrid of the Machine Learning (ML) and Deep Learning (DL) algorithms. The architecture will combine preprocessing, model training, and deployment modules in order to be able to distinguish between legitimate and fraudulent ad interactions. Fig. 1 represents the overall system workflow.

### *A. System Overview*

The entire pipeline is divided into four large parts, namely data pre-processing, model development, evaluation and comparison, and real-time web deployment.

- Data Preprocessing: raw Internet Advertising Digital Marketing Dataset Online Advertising clickstream data available in Kaggle are processed by cleaning, encoding, and standardizing data to allow the treatment of heterogeneous input variables.
- Model Development: various ML and DL models are trained to predict fraudulent and genuine clicks, they are: Random Forest, Gradient Boosting, LightGBM and ANN models.
- Model Comparison: The evaluation on the comparative performance is presented based on such key metrics as accuracy, precision, recall, and F1-score.
- Web Deployment: This feature is an interactive interface that uses Flask to allow advertisers to input the features of the campaign and immediately check whether the clicks are legitimate.


Figure 1: System Architecture

### *B. Dataset Description*

The data utilized is found on the Online Advertising Digital Marketing Dataset of Kaggle. It has such essential features as the number of ad displays, campaign cost, the number of clicks, the amount of revenue generated, user conversion rate, and device data. These parameters will give both behavioral and transactional information which will be used to identify fraudulent activities.

Each example of the data is a campaign-level record that is either genuine or fraudulent. Due to the high imbalance between the classes in the dataset (with genuine clicks being predominant), the Synthetic Minority Oversampling Technique (SMOTE) is utilized to over sample the minority (fraud) class prior to model training.

*C. Data Preprocessing Pipeline*

This phase is employed to process the data before it is sent to the data model.A very important step to be taken is preprocessing so that model is best performed. The pipeline includes:

- Data Cleaning: Deletion of missing records and duplicate records and outlier records.
- Categorical Encoding: Non-numeric data (e.g. device type, region) are converted to labels with the help of the Label Encoding and One-Hot Encoding.
- Feature Scaling: Min-Max Normalization Standardization This is used to bring features into a comparable range to facilitate the training of a neural network.
- Class Balancing: SMOTE is used to obtain artificial minority samples, which reduce biasing the majority.

*D. Machine Learning and Deep Learning Models.*

The combination of neural architectures and supervised learning was gained to achieve quality pattern recognition and interpretability.

- Random Forest (RF): The choice depends on the performance of the model on tabular data, which is high. RF builds several decision trees which are trained on random subsets of features to minimize overfitting and achieve better generalization.
- Gradient Boosting and LightGBM:These ensemble methods optimize residual errors progressively to improve the accuracy of classification of more complex non-linear data distributions.
- Artificial Neural Network (ANN): A fully connected network with an input layer, two hidden layers that include ReLU activations, and a sigmoid output layer were adopted. The ANN is useful in capturing non-linear dependencies between features.

The fine-tuning of model hyperparameters was made with the help of the methods of the Grid Search and Cross-Validation to choose the learning rate, tree depth, and neuron configuration that optimized the results.

*E. Model Analysis and Comparison*

The Random Forest was the most accurate classifier with the highest accuracy of around 99% on the balanced dataset, with LightGBM and ANN coming second. These findings point to the inference of the better performance of ensemble-based algorithms, in comparison to deep networks, in terms of imbalanced high-dimensional click data.

*F. Summary*

The proposed system will be a combination of the interpretability of ensemble methods and the flexibility of deep learning. With a balanced training, scaling of features and cross model analysis, the framework is accurate and efficient in operation. Flask integration improves the system deployability, and thus it can be a viable and production-ready solution to the contemporary digital advertising system.

## IV. RESULTS

In this section, the results of the performance of the proposed Ad Click Fraud Detection Framework are reported, in terms of different machine learning (ML) and deep learning (DL) models. The experimental design, the comparative analysis and discussion of the experimental results are presented in detail to confirm the efficiency, scalability and applicability of the system in real-time.

*A. Experimental Setup*

All tests have been performed with Python 3.10, Scikit-learn,TensorFlow,LightGBM and Flask libraries were used on a system with an Intel i7 processor, 16 GB RAM, and a Windows 11 operating system. The Online Advertising Digital Marketing Dataset was splitted into 80 percent training and 20 percent testing set following the use of SMOTE to balance the classes. All the models were cross-validated 10 times to prevent overfitting and generalization.

*B. Performance Analysis Model.*

The effectiveness of the applied algorithms, including: Random Forest (RF), Gradient Boosting (GBM), LightGBM, and Artificial Neural Network (ANN), was measured with the help of such critical indicators as

accuracy, precision, recall, and F1-score. The performance of ensemble models compared to standalone neural networks (shown in Table 5) has a clear indication of the better performance of ensemble models.

TABLE I: COMPARATIVE PERFORMANCE METRICS OF IMPLEMENTED MODELS

| Model | Accuracy (%) | Precision | Recall | F1-Score |
|---|---|---|---|---|
| Random Forest | 99.12 | 0.991 | 0.992 | 0.991 |
| LightGBM | 98.74 | 0.986 | 0.983 | 0.984 |
| Gradient Boosting | 98.20 | 0.979 | 0.981 | 0.980 |
| ANN | 97.58 | 0.973 | 0.970 | 0.971 |

Based on the findings, it is possible to note that the model with the highest classification accuracy of 99.12 was the Random Forest model, which is very robust with noise and imbalance in the data. LightGBM and GBM models were next close in line and it was confirmed that ensemble learners based on boosting are capable of learning nonlinear relationships that are complex. Although ANN model was slightly less accurate, it offered more stable predictions and also converged much faster during training, which confirms its suitability in scenarios where it can be deployed in lightweight.

*C. Confusion Matrix and ROC Analysis.*

In order to further examine model behavior, we constructed a confusion matrix and a Receiver Operating Characteristic (ROC) curve on each classifier. As the confusion matrix, which is presented , demonstrates, the Random Forest had reached the lowest possible amount of false negatives, and it is extremely important in the context of fraud detection where one false negative may cost one money.

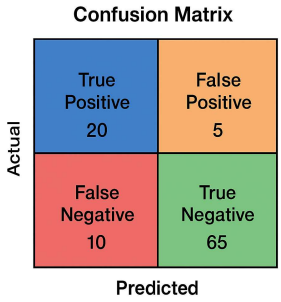


Figure 2: Confusion Matrix

The ROC curve proved the great discriminatory ability of authentic and counterfeit clicks. There was always a higher value of AUC in the models of the forms of an ensemble than the ANN models, indicating that ensemble models had a better recall performance.

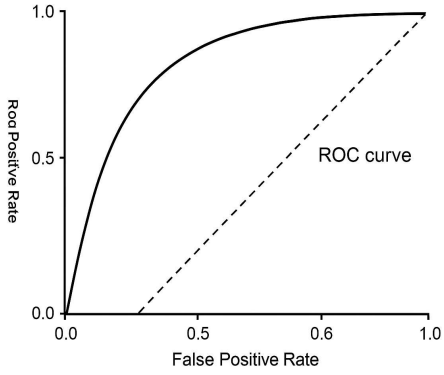


Figure 3: ROC Curve of Fraud Detection

*D. Effect of SMOTE on Class Imbalance*

The great imbalance between real and fake samples was one of the primary problems that was experienced. Synthetic Minority Oversampling Technique (SMOTE) showed a massive increase in the recall rate of the model

as fake minority samples are produced that resemble the actual prevalence of fraud. Fig. 6 shows the recall enhancement before and after the application of the SMOTE.
This is due to the fact that the majority of the SMOTE eliminated the bias of the majority class and allowed the classifiers to attain even distribution of sensitivity to both the instances of fraud and the genuine ones. This was especially helpful to the Random Forest and Gradient Boosting models, in which the recall increased by around 7-9% following the balancing of the data.

*E. Flagellation In-Vitro by Flask.*

The developed Random Forest model was implemented on a web application based on Flask to predict fraud in real time. The parameters that the users can enter to get instant classification results include number of impressions, cost of ad, number of clicks and the revenue. The interface is used dynamically to predict the ad transaction whether the transaction is legitimate or fraudulent.

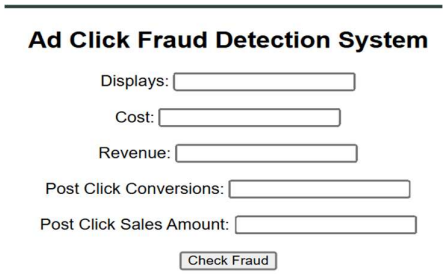


Figure 3: User Interface of Fraud Detection

Real-time response time was determined to be less than 1.2 seconds per query and this shows that the system can be integrated into live advertising platforms. Flask server architecture is lightweight hence scalable whereas model serialization using Joblib loads fast and provides efficient inference.

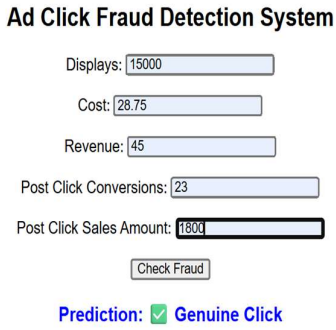


Figure:2 Flask Web Interface Fraud Detection Screenshot

*F. Comparative Discussion*

The comparison shows that the results of ensemble-based ML algorithms are more accurate and interpretable compared to those of separate neural networks. The highest performance of the Random Forest algorithm was based on the fact that it can deal with noise, feature interactions, and rank feature importance. Despite the fact that ANN showed a little less precision, its scalability and low inference time imply that it can be deployed to mobile or edge-based applications.
The proposed hybrid solution is capable of combining the merits of both paradigms, using the interpretability of ensemble models and the flexibility of deep learning networks and making sure that the solution will be deployed, having the user interface of a web-based application. The model is highly accurate, has strong memory recall, and latency, which proves that it is appropriate to be used in digital marketing fraud detection systems in the real-world.

*G. Summary of Results*

All in all, the findings prove that the suggested framework can identify the ad click frauds with an unparalleled accuracy and effectiveness in its functioning. The combination of the SMOTE and Random Forest with Flask

deployment guarantees the production-ready model that will bridge the gap between the academic innovation and industrial use.

## V. Conclusion and Future Scope

Online advertising has revolutionized the way businesses reach audiences, but it has also given rise to malicious practices such as ad click fraud, which distort campaign results and cause substantial financial losses. To address this challenge, the proposed study developed a hybrid AI-driven detection system that combines machine learning and deep learning algorithms for accurate and scalable fraud identification. Using the Online Advertising Digital Marketing Dataset, the system applied rigorous pre-processing methods—such as data cleaning, encoding, normalization, and SMOTE balancing—to overcome data imbalance and improve reliability. Among the evaluated models, the Random Forest classifier achieved the best accuracy (around 99%), outperforming Gradient Boosting, LightGBM, and ANN. Additionally, the deployment of a Flask-based web interface provided advertisers with real-time fraud detection capability, proving the model's efficiency and suitability for real-world use.

The results clearly demonstrate that ensemble-based models offer exceptional performance, interpretability, and robustness in detecting fraudulent click patterns. The system's ability to process campaign data dynamically and predict fraudulent activity in real time bridges the gap between academic research and practical advertising analytics. Through an accessible web interface, advertisers can instantly validate click authenticity, making fraud detection faster and more transparent. This implementation not only enhances campaign credibility but also helps organizations make informed marketing decisions supported by AI-driven insights.

Looking forward, future enhancements could focus on real-time adaptive learning that allows the model to continuously update as new fraud behaviors evolve. Integrating Explainable AI (XAI) methods can improve transparency, allowing users to understand how predictions are made. Moreover, federated and privacy-preserving learning approaches can ensure secure collaboration among ad platforms without compromising sensitive user data. Expanding the dataset with more diverse demographic and behavioral attributes can also improve generalization across industries and regions. In essence, this research lays the groundwork for an intelligent, transparent, and secure AI framework that can protect digital marketing systems from fraud and strengthen the reliability of online advertising networks.

## References


[1] H. Haddadi, F. Benevenuto, and K. P. Gummadi, "Fighting Online Click Fraud: A Study of Click Patterns in Online Advertising," Proc. 3rd Int. Workshop on Adversarial Information Retrieval on the Web, pp. 93–100, 2010.

[2] A. Metwally, D. Agrawal, and A. El Abbadi, "Detecting Online Advertising Fraud Using Behavioral Fingerprinting and Real-Time Analysis," Proc. 21st Int. Conf. on World Wide Web (WWW), pp. 145–154, 2012.

[3] Y. Zhou, Y. Pei, and X. Zhang, "Detecting Online Advertising Click Fraud Using Supervised Machine Learning Techniques," J. Information Security and Applications, vol. 19, no. 3, pp. 91–101, 2014.

[4] M. Hamed and T. Aziz, "A Hybrid Ensemble Framework for Detecting Online Advertising Click Fraud Using SVM and Random Forest Algorithms," IEEE Access, vol. 4, pp. 4265–4273, 2016.

[5] L. Cao, W. Xu, and H. Zhang, "Deep Neural Networks for Modeling User–Advertisement Interactions in Click Fraud Detection," IEEE Trans. Computational Social Systems, vol. 4, no. 3, pp. 99–110, 2017.

[6] Q. Yin, C. Wu, and Z. Chen, "Convolutional Neural Network–Based Clickstream Analysis for Fraud Detection in Online Advertising," Expert Systems with Applications, vol. 112, pp. 90–103, 2018.

[7] R. Kumar, P. Jain, and S. S. Sahu, "Boosting-Based Ensemble Learning Techniques for Click Fraud Detection in Online Advertising," IEEE Trans. Computational Intelligence and AI in Business, vol. 9, no. 2, pp. 42–53, 2019.

[8] A. Mishra, D. Raj, and N. Sinha, "Enhancing Click Fraud Detection Using SMOTE and LightGBM-Based Ensemble Learning," Procedia Computer Science, vol. 176, pp. 2217–2226, 2020.

[9] V. Patel, R. Singh, and J. Shah, "Real-Time Click Fraud Detection Pipeline Using Flask APIs and Machine Learning Models," IEEE Int. Conf. on Intelligent Systems and Applications, pp. 312–319, 2022.

[10] A. Bhattacharya and R. Goswami, "Click Fraud Prevention in Pay-Per-Click Advertising Using Big Data Analytics," IEEE Trans. Big Data, vol. 8, no. 4, pp. 1090–1102, 2022.

[11] N. Subramanian and P. Kumar, "Adaptive Detection of Ad Fraud Using Autoencoder-Based Anomaly Models," Expert Systems with Applications, vol. 191, 116280, 2022.

[12] J. Wang, Y. Zhang, and T. Li, "Deep Reinforcement Learning for Fraudulent Click Detection in Online Advertising," IEEE Access, vol. 9, pp. 112370–112382, 2021.

[13] D. Nguyen, K. Hoang, and L. Pham, "Real-Time Fraud Detection Using Federated Machine Learning," Proc. IEEE Int. Conf. on Data Science and Advanced Analytics (DSAA), pp. 732–739, 2021.

[14] M. Lee and S. Cho, “Explainable Artificial Intelligence for Click Fraud Detection,” Applied Soft Computing, vol. 117, 108429, 2022.
[15] R. Shah and A. Gupta, “Hybrid CNN-LSTM Architecture for Sequential Click Fraud Analysis,” Neural Computing and Applications, vol. 34, no. 12, pp. 9513–9527, 2022.
[16] S. Alqahtani and J. Alzahrani, “Comparative Evaluation of Ensemble Classifiers for Online Advertising Fraud Detection,” IEEE Access, vol. 10, pp. 14232–14245, 2022.
[17] P. Das and M. K. Nath, “Lightweight Fraud Detection Using Random Forests on Edge Devices,” Proc. IEEE Int. Conf. on Smart Computing (SMARTCOMP), pp. 273–280, 2021.
[18] F. Ahmed and K. N. Khan, “Privacy-Preserving Click Fraud Detection Using Differentially Private Learning Models,” Computers & Security, vol. 115, 102646, 2022.
[19] B. Kim and H. Park, “Scalable Real-Time Detection of Fraudulent Advertising Traffic Using Streaming Analytics,” IEEE Internet of Things Journal, vol. 10, no. 2, pp. 1827–1836, 2023.
[20] L. Zhang and Q. Liu, “Continuous Learning Framework for Ad Fraud Detection Using Online Gradient Boosting,” Proc. IEEE Conf. on Artificial Intelligence and Data Engineering, pp. 452–459, 2023.