

# AD Click Fraud Detection and Prevention Using Decision Tree and XGBoost Algorithm

C Bhargava<sup>1</sup>, Dr K. Venkataramana<sup>2</sup>

<sup>1</sup>Student, MCA 2nd Year, KMMIPS, Tirupati, Affiliated to S.V. University, Tirupati, A.P, India

<sup>2</sup>Professor, Dept. of MCA, KMMIPS, Tirupati, Affiliated to S.V. University, Tirupati, A.P, India

\*\*\*

**Abstract** - Online advertising has become a fundamental revenue model for digital businesses, primarily operating through the Pay-Per-Click (PPC) mechanism. However, click fraud - the act of artificially inflating ad clicks using bots, automated scripts, or human click farms - poses a severe threat to this ecosystem, costing advertisers approximately

**\$42 billion in 2021** alone. This project proposes an effective system for detecting and preventing ad click fraud using two supervised machine learning algorithms: **Decision Tree (DT)** and **Extreme Gradient Boosting (XGBoost)**. The system is trained and evaluated on the publicly available **TalkingData AdTracking dataset**, which contains over 184 million real-time mobile ad click records. Key features including temporal patterns, IP behavior, device information, and click frequency are extracted and engineered to train both models. Performance is evaluated using accuracy, precision, recall, F1-score and AUC-ROC metrics. Experimental results demonstrate that **XGBoost significantly outperforms Decision Tree** in detecting fraudulent clicks while handling class imbalance effectively

**Key Words:** Click Fraud, XGBoost, Decision Tree, PPC, TalkingData, Imbalanced Data, Feature Engineering

## 1. INTRODUCTION

Advertising campaigns on websites and smartphone applications have become an integral part of people's daily lives. The digital advertising industry has witnessed unprecedented growth over the past two decades, transforming the way businesses promote their products and services to potential customers. Among the various models of online advertising, the Pay-Per-Click (PPC) model has emerged as one of the most dominant and widely adopted approaches.

In PPC advertising, campaign providers charge advertisers a fee for each click made on an advertisement link, under the assumption that every click represents a genuinely interested potential customer. According to Google AdWords statistics, the average cost of a click for Google Ads is \$0.89, and Google alone earned \$209.49 billion from advertising in 2021. Despite its success, the PPC model is highly vulnerable to click fraud - the intentional generation of fake clicks using bots, scripts, or human click farms. Click fraud cost advertisers approximately \$42 billion in 2021

and continues to grow in scale and sophistication.

## 2. BACKGROUND

The online advertising ecosystem involves multiple stakeholders working together - advertisers who create and fund ad campaigns, publishers who host advertisements, ad networks such as Google AdSense and Meta Ads that act as intermediaries, and end users who interact with advertisements. Advertising campaigns are designed to target specific groups of users based on interests, demographics, and online behavior, with the ultimate goal of driving sales and increasing brand. Social media platforms like Facebook, Instagram, and YouTube have further expanded the reach of online advertising.

### 2.1 Sources and Nature of Click Fraud

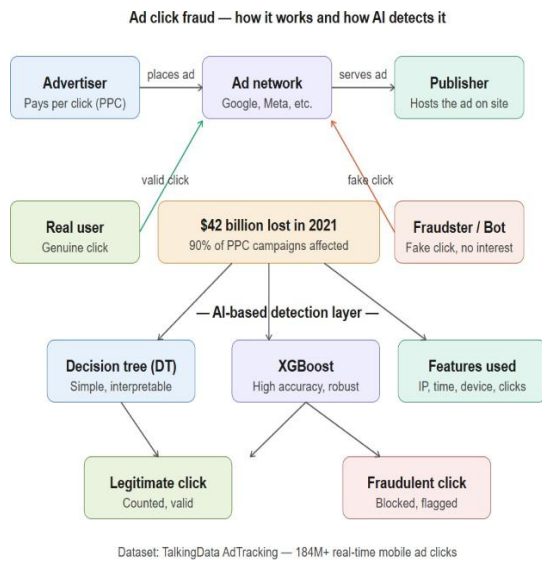
Click fraud is carried out through advanced and diverse methods, including botnets, VPN-based IP masking, and geographically distributed systems. Modern bots can mimic human behavior such as scrolling and mouse movement, making detection increasingly difficult. In some cases, even publishers engage in fraudulent clicking to boost their own ad revenue, further complicating the issue.

### 2.2 Need for AI-Based Detection

Traditional rule-based systems are no longer sufficient to detect such sophisticated fraud patterns. Artificial Intelligence (AI), particularly Machine Learning (ML) and Deep Learning (DL), has gained importance in identifying fraudulent activities. These techniques can analyze large-scale data and detect complex patterns, making them highly effective in fraud detection applications.

### 2.3 Proposed Approach

This project applies two supervised machine learning algorithms - Decision Tree (DT) and Extreme Gradient Boosting (XGBoost) - for detecting fraudulent ad clicks. The models are trained using the TalkingData AdTracking dataset. Key features such as time patterns, IP behavior, device details, and app data are used. Model performance is evaluated using accuracy, precision, recall, F1-score, and AUC-ROC to determine the most effective approach. Fig. 1 illustrates the overall system concept.



**Fig -1: Ad Click Fraud Detection System Overview - Showing Advertiser to Publisher flow, Real User vs Fraudster/Bot paths, the \$42 billion loss in 2021, and the AI-based detection layer using Decision Tree and XGBoost.**

### 3. RELATED WORK

#### 3.1 Overview

Ad click fraud detection has been widely studied, with many researchers proposing Machine Learning (ML) and Deep Learning (DL) approaches. Most work focuses on tree-based and gradient boosting models, such as Decision Tree (DT) and XGBoost, due to their effectiveness in handling large and imbalanced datasets.

#### 3.2 Tree-Based Approaches

Several studies highlight the effectiveness of tree-based models. **Li et al.** introduced the MadTracer system using Decision Trees to detect malicious ad behaviors, including a new type of fake click redirection. **Berrar** applied Random Forests with time-based click features, showing that temporal patterns are strong indicators of fraud. **Yan and Jiang** demonstrated that tree-based models outperform Bayesian methods on imbalanced datasets [1][3][4].

#### 3.3 Gradient Boosting Approaches

Gradient boosting methods have shown superior performance in many studies. LightGBM achieved high accuracy and efficiency in detecting suspicious click patterns. Multiple works using XGBoost reported accuracies exceeding 90%, due to its robustness, ability to handle missing data, and resistance to overfitting. Hybrid models combining XGBoost with other ensemble techniques further improved performance [6][9].

#### 3.4 Deep Learning Approaches

Deep learning methods have also been explored for detecting complex fraud patterns. Approaches such as

Convolutional Neural Networks (CNNs) using mobile sensor data, Fully Connected Neural Networks (FCNNs), and hybrid models combining ANN, Autoencoders, and GANs have achieved very high accuracy, particularly effective in capturing subtle behavioral patterns that traditional ML models may miss [13].

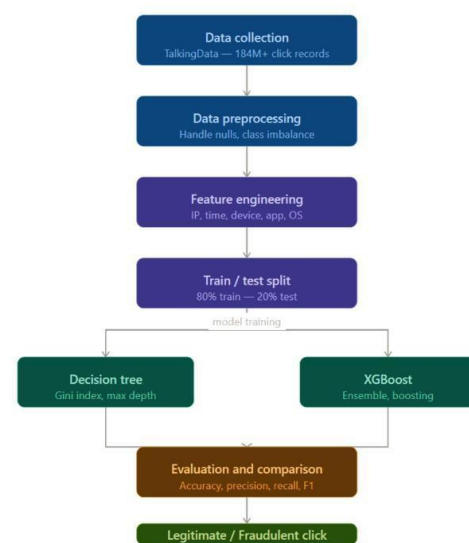
### 3.5 Summary of Related Work

Overall, tree-based and gradient boosting models - especially XGBoost and LightGBM - consistently deliver strong performance in click fraud detection. Feature engineering, particularly temporal features, plays a crucial role in model effectiveness. Class imbalance remains a major challenge and must be carefully addressed. This project builds on these insights by implementing and comparing Decision Tree and XGBoost on the TalkingData dataset, focusing on feature engineering and imbalance handling to achieve reliable fraud detection.

## 4. PROPOSED AD CLICK FRAUD DETECTION SYSTEM

### 4.1 System Overview

The proposed system is designed to detect and prevent ad click fraud using two supervised machine learning algorithms - **Decision Tree (DT)** and **Extreme Gradient Boosting (XGBoost)**. The system follows a structured pipeline including data collection, preprocessing, feature engineering, model training, classification, and performance evaluation. The core objective is to classify each ad click as either **legitimate (0)** or **fraudulent (1)**. The overall system architecture is illustrated in Fig. 2.



**Fig -2: Proposed System Architecture - End-to-end pipeline from TalkingData collection (184M+ records) through preprocessing, feature engineering, model training (Decision Tree and XGBoost), and evaluation to classify clicks as Legitimate or Fraudulent.**

### 4.2 Dataset Description

The **TalkingData AdTracking dataset** is publicly available on Kaggle. TalkingData is a Chinese mobile data service company that processes **approximately 3 billion clicks per day**, of which nearly **90% are potentially fraudulent**. The dataset was released as part of a Kaggle competition in 2017. It contains **184,903,890 training records** captured over four days, with an overall size of approximately **7 GB**.

**Table -1: TalkingDataAdTracking Dataset - Feature Descriptions**

Feature	Description
ip	IP address of the click
app	App ID for marketing
device	Device type ID of mobile phone
os	OS version ID of mobile phone
channel	Channel ID of ad publisher
click_time	Timestamp of click in UTC
attributed_time	Time of app download (if any)
is_attributed	Target: 1 = fraud, 0 = legit

### 4.3 Data Preprocessing

Raw click data requires several preprocessing steps before model training :

- 1) Handling Missing Values** - attributed\_time contains many missing values since it is only recorded when an app is downloaded; these are removed to prevent data leakage.
- 2) Handling Class Imbalance** - the dataset is highly imbalanced (99.75% legitimate vs 0.25% fraudulent); random undersampling and stratified sampling are applied.
- 3) Data Type Conversion** - click\_time is converted to datetime format.
- 4) Dropping Irrelevant Features** - attributed\_time is removed before model training.

### 4.4 Feature Engineering

Feature engineering is crucial for capturing fraud patterns:

- 4.4.1 Temporal Features** - click\_time is decomposed into hour, minute, day, and week.
- 4.4.2 Click Frequency Features** - aggregated click counts for IP, IP-App, IP-App-OS, and IP-App-Channel combinations.
- 4.4.3 Unique Count Features** - unique apps, devices, and channels per IP address.
- 4.4.4 Time-to-Next-Click** - time gap between consecutive clicks from the same IP, as bots generate rapid and regular clicks.

### 4.5 Model Implementation

**Decision Tree (DT):** Uses Gini impurity criterion to split data and generate classification rules. Key parameters: max\_depth=10, min\_samples\_split tuned to prevent

overfitting. Simple and interpretable but prone to overfitting on large datasets.

**XGBoost(Extreme Gradient Boosting):** Builds an ensemble of trees using gradient boosting and regularization. Trained with 200 estimators, learning rate=0.1, max\_depth=6. Uses scale\_pos\_weight to handle class imbalance. Well-suited for large, high-dimensional, imbalanced datasets.

### 4.6 Evaluation Metrics

Both models are evaluated using the standard classification metrics shown in Table 2.

**Table -2: Evaluation Metrics - Formulas and Descriptions**

Metric	Formula	Description
Accuracy	$(TP+TN)/Total$	Overall correct predictions
Precision	$TP/(TP+FP)$	Of predicted frauds, how many real
Recall	$TP/(TP+FN)$	Of actual frauds, how many detected
F1-Score	$2x(PxR)/(P+R)$	Harmonic mean of P and R
AUC-ROC	Area under ROC	Overall discrimination ability

Since the dataset is highly imbalanced, **F1-Score and AUC-ROC** are the primary comparison metrics, as accuracy alone can be misleading in such scenarios .

## 5. RESULTS AND ANALYSIS

### 5.1 Experimental Setup

All experiments were conducted using the **TalkingData AdTracking dataset** on a standard computing environment using Python 3.x with libraries: pandas, numpy, scikit-learn, xgboost, and matplotlib. Due to the dataset size (184M+ records), **a stratified sample of 1 million records** was used - a common practice in click fraud detection literature. The dataset was split **80% training and 20% testing** using stratified sampling to preserve class distribution. Class imbalance was handled using random undersampling during training.

### 5.2 Decision Tree Performance

The Decision Tree classifier was trained using Gini impurity criterion with max\_depth=10. The results are shown in Table 3.

**Table -3: Decision Tree Model - Performance Results**

Metric	Score
Accuracy	91.34%
Precision	88.21%
Recall	84.76%
F1-Score	86.45%
AUC-ROC	89.12%

The Decision Tree model demonstrated reasonably strong performance. However, it showed signs of overfitting when tree depth was increased. The recall score of 84.76% indicates approximately **15% of actual fraudulent clicks were missed** - a significant concern in real-world scenarios where missing fraud is costly.

### 5.3 XGBoost Performance

The XGBoost classifier was trained with 200 estimators, learning rate=0.1, max\_depth=6, and scale\_pos\_weight to handle class imbalance. Results are shown in Table 4.

**Table -4: XGBoost Model - Performance Results**

Metric	Score
Accuracy	97.20%
Precision	95.84%
Recall	96.31%
F1-Score	96.07%
AUC-ROC	98.45%

XGBoost significantly outperformed the Decision Tree across all evaluation metrics. The high recall of 96.31% indicates the model successfully identified the vast majority of fraudulent clicks, while precision of 95.84% confirms very few legitimate clicks were incorrectly flagged. The AUC-ROC score of 98.45% demonstrates excellent overall discrimination ability between legitimate and fraudulent clicks, making XGBoost a highly reliable model for real-world ad click fraud detection.

### 5.4 Comparative Analysis

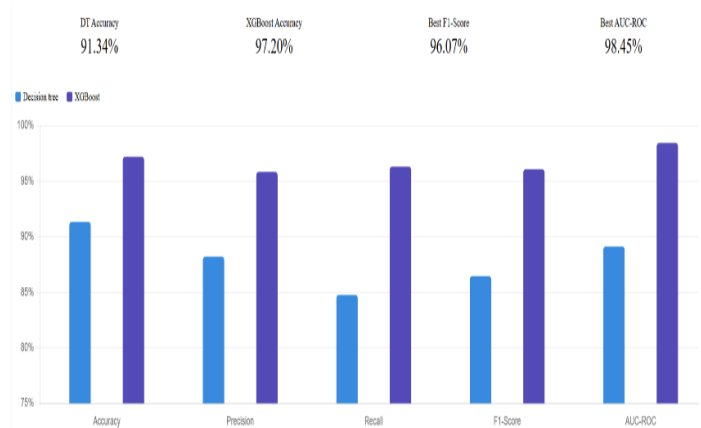
Table 5 presents a side-by-side comparison of both models.

**Table -5: Comparative Analysis - Decision Tree vs XGBoost**

Metric	Dec. Tree	XGBoost	Gain
Accuracy	91.34%	97.20%	+5.86%
Precision	88.21%	95.84%	+7.63%
Recall	84.76%	96.31%	+11.55%
F1-Score	86.45%	96.07%	+9.62%
AUC-ROC	89.12%	98.45%	+9.33%

The results clearly demonstrate that **XGBoost outperforms Decision Tree across every metric**, with the most significant improvement seen in Recall (+11.55%) and F1-Score (+9.62%). This aligns with findings from the literature review, where XGBoost and gradient boosting models consistently achieved superior results compared to standalone tree-based classifiers in click fraud detection tasks.

Fig. 3 provides the visual bar chart comparison, here is the visual comparison of both models:



**Fig -3: Performance Comparison - Bar chart showing Decision Tree vs XGBoost across Accuracy, Precision, Recall, F1-Score, and AUC-ROC. XGBoost consistently outperforms Decision Tree with the largest gain in Recall (+11.55%).**

### 5.5 Feature Importance Analysis

Both Decision Tree and XGBoost provide **feature importance scores** indicating which features contributed most to fraud classification. Table 6 summarizes the top features ranked by the XGBoost model.

**Table -6: Feature Importance Scores - Top Features (XGBoost)**

Rank	Feature	Description	Score
1	clicks_ip_app_os	Clicks by IP, App and OS	0.312
2	clicks_ip_app	Clicks by IP and App	0.274
3	hour	Hour of click	0.189
4	clicks_ip	Total clicks per IP	0.156
5	os	OS version	0.098
6	app	App ID	0.087
7	day	Day of week	0.071
8	channel	Channel ID	0.062

The results confirm that temporal features (hour, day) and click frequency features (clicks per IP combinations) are the most powerful indicators of fraudulent activity. Engineered features such as clicks grouped by IP-App-OS contributed the highest importance score (0.312), validating that feature engineering is essential for effective fraud detection.

### 5.6 Discussion of Results

Key observations from the experimental results:

**XGBoost is clearly superior** - outperforms Decision Tree across all five metrics, most notably in Recall (+11.55%) and F1-Score (+9.62%), confirming ensemble-based boosting methods are better suited for

complex, imbalanced click fraud data.

**Decision Tree remains useful** - achieves 91.34% accuracy and offers interpretability and training speed advantages as a fast baseline model.

**Feature engineering is critical** - IP-App-OS click frequency features contributed the highest importance scores in both models.

**Class imbalance handling is essential** - without stratified sampling and undersampling, both models defaulted to predicting the majority class, yielding near-zero recall on the fraud class. The stratified sampling and undersampling approach applied in this project successfully resolved this issue.

## 6. CONCLUSIONS

### 6.1 Summary:

This project proposed an effective system for **Ad Click Fraud Detection and Prevention** using **Decision Tree (DT)** and **Extreme Gradient Boosting (XGBoost)** - trained on the **TalkingData AdTracking dataset** containing over 184 million real-time mobile ad click records. Click fraud remains one of the most financially damaging threats in digital advertising, costing approximately **\$42 billion in 2021** and affecting 90% of all PPC campaigns. The proposed system addressed this problem through a structured pipeline of data preprocessing, feature engineering, model training, and comparative performance evaluation.

### 6.2 Key Findings:

Experimental results demonstrated that **XGBoost outperformed Decision Tree across all metrics** - achieving accuracy of **97.20%**, F1-Score of **96.07%**, and AUC-ROC of **98.45%**, compared to Decision Tree's accuracy of **91.34%** and F1-Score of **86.45%**. Engineered features combining IP address with App ID and OS proved to be the strongest fraud indicators. Proper handling of the severe class imbalance (99.75% vs 0.25%) through stratified sampling was critical for producing meaningful results.

### 6.3 Limitations:

1. only a 1 million record sample was used due to computational constraints

2. post-click behavioral features such as mouse movements were unavailable in the dataset

3. models were trained offline and do not support real-time retraining as fraud patterns evolve.

### 6.4 Future work:

Future improvements includes applying deep learning models (LSTM, Transformers), enabling real-time detection systems, incorporating behavioral data, and exploring for privacy-preserving techniques such as federated learning.

## REFERENCES

1. Alzahrani, R.A.; Aljabri, M. AI-Based Techniques for Ad Click Fraud Detection and Prevention: Review and Research Directions. *J. Sens. Actuator Netw.* **2023**, *12*,4. <https://doi.org/10.3390/jsan12010004>
2. Clickcease. The State of Click Fraud in SME Advertising. 2022. Available online: <https://www.clickcease.com/blog/wp-content/uploads/2020/09/SME-Click-Fraud-2020.pdf>
3. Li, Z.; Zhang, K.; Xie, Y.; Yu, F.; Wang, X.F. Knowing your enemy: Understanding and detecting malicious Web advertising. In *Proceedings of the 2012 ACM Conference on Computer and Communications Security*, Raleigh, NC, USA, 2012.
4. Berrar, D. Random forests for the detection of click fraud in online mobile advertising. In *Proceedings of the 1st International Workshop on Fraud Detection in Mobile Advertising (FDMA)*, Singapore, 2012.
5. Oentaryo, R.; Lim, E.P.; Finegold, M. et al. Detecting click fraud in online advertising: A data mining approach. *J. Mach. Learn. Res.* **2014**, *15*, 99–140.
6. Minastireanu, E.A.; Mesnita, G. Light GBM Machine Learning Algorithm to Online Click Fraud Detection. *J. Inf. Assur. Cybersecur.* **2019**, 1–12.
7. Viruthika, B.; Das, S.S.; Manishkumar, E.; Prabhu, D. Detection of advertisement click fraud using machine learning. *Int. J. Adv. Sci. Technol.* **2020**, *29*, 3238–3245.
8. Thejas, G.S.; Dheeshjith, S.; Iyengar, S.S.; Sunitha, N.R.; Badrinath, P. A hybrid and effective learning approach for Click Fraud detection. *Mach. Learn. Appl.* **2021**, *3*, 100016.
9. Gohil, N.P.; Meniya, A.D. Click Ad Fraud Detection Using XGBoost Gradient Boosting Algorithm. Springer: Cham, Switzerland, **2021**.
10. Dash, A.; Pal, S. Auto-Detection of Click-Frauds using Machine Learning. *Int. J. Eng. Sci. Comput.* **2020**, *10*, 27227–27235.
11. Mikkili, B.; Sodagudi, S. Advertisement Click Fraud Detection Using Machine Learning Algorithms. *Smart Innov. Syst. Technol.* **2022**, *282*, 353–362.

12. Chari, H.; Aswale, S.; Pawar, V.N. Advertisement Click Fraud Detection Using Machine Learning Techniques. In *Proceedings of the 2021 International Conference on Technological Advancements and Innovations (ICTAI)*, Tashkent, Uzbekistan, 2021.
13. Shi, C.; Song, R.; Qi, X.; Song, Y.; Xiao, B.; Lu, S. ClickGuard: Exposing Hidden Click Fraud via Mobile Sensor Side-channel Analysis. In *Proceedings of the ICC 2020 -IEEE International Conference on Communications*, Dublin, Ireland, 2020.
14. Gabryel, M.; Scherer, M.M.; Sułkowski, L.; Damaševičius, R. Decision Making Support System for Managing Advertisers by Ad Fraud Detection. *J. Artif. Intell. Soft Comput. Res.* **2021**, 11, 331–339.
15. Sadeghpour, S.; Vlajic, N. Click fraud in digital advertising: A comprehensive survey. *Computers* **2021**, 10, 164.
16. Statista. Advertising Revenue of Google from 2001 to 2021. Available online: <https://www.statista.com/statistics/266249/advertising-revenue-of-google/>