Identifying Auction Fraud with Decision Tree-Based Methods

Theodor P. Teske
University of California, Los Angeles
Los Angeles, CA
teddyteske@ucla.edu

***Abstract*—Shill Bidding (SB) is a pernicious form of auction fraud that is particularly difficult to detect. We employ Random Forests (RF), Stochastic Gradient Boosting (GBM), and XGBoost (XGB) to discriminate between instances of shill bidding and non-fraudulent bidding in a dataset mined from eBay. We compare two methods to address a large class imbalance in the dataset: one involving class weighting, and another using SMOTE. Uniquely, our methods also allow us to determine which factors are most predictive of SB, and they outperform all other models focused on SB detection in the literature.**

Keywords—Shill Bidding, Random Forests, Gradient Boosting, SMOTE, Class Imbalance, Online Fraud Detection

I. INTRODUCTION

The online auction has emerged as a popular business model in the largely online modern retail space. Business to consumer (B2C) ecommerce sales alone are forecasted to achieve a nearly 22% share of total global retail sales in 2024 [1]. The most recognizable name in the online auction space, eBay, had 133 million active users as of the first quarter of 2023 [2]. It is therefore unsurprising that online auction fraud accounts for the largest part of all internet fraud. The Internet Fraud Complaint Center (IFCC) classifies auction fraud into six categories: non-delivery of goods, misrepresentation of the items, triangulation, fee staking, selling of black-market goods, multiple bidding and shill bidding [3]. This paper will focus on *shill bidding* (SB), a form of auction fraud in which a seller enters bids on their own item or colludes with one or more bidders in the auction to place bids on their behalf. The purpose of shill bidding is to artificially inflate the selling price of the item, so that legitimate bidders are forced to pay more to win. Alternatively, the seller might create multiple fake bidding accounts to submit bids, in what is known as *multiple bidding*, with the same goal [4]. Of all types of auction fraud, shill bidding and multiple bidding are the most difficult to discourage for two reasons. First, they leave no direct evidence, unlike other more concrete forms of fraud such as non-delivery of goods. Second, it is difficult to precisely measure the financial loss associated with multiple bidding and shill bidding, making it difficult to punish fraudsters via law enforcement [3]. What's more, online auctions are particularly susceptible to shill bidding as they allow for a high level of anonymity; indeed, many only require an email address to create an account. The relative ease of creating accounts is appealing to would-be fraudsters, and the anonymity means it is hard to prove that someone is guilty of shill bidding [5]. As buyers are forced to overpay, especially on high-priced items, online auctions may lose their credibility [6]. These circumstances motivate an investigation into the best ways to algorithmically detect and ultimately prevent shill bidding in online auctions.

II. RELATED WORK

Ford, Xu, and Valova [7] represent an early effort to identify SB in an online auction setting, looking at eBay auctions involving the sales of a "Used Playstation 3." They first implement a hierarchical clustering algorithm, then employ an

Artificial Neural Network (ANN) to explore the bidding behavior within each cluster. However, an ANN is computationally costly and may fall victim to local minima, and the authors classify bidders based on their behavior over many auctions. Thus, there is no way to determine whether SB is present in a particular auction using this method.

Gupta and Mundra in [8] employ a Hidden Markov Model (HMM) to detect auction fraud using two parameters, the number of bids and the bid values. The Markov model is split into a training layer and a detection layer; in the former, K-means clustering is used to generate an initial set of probabilities based on bidding behavior, while in the latter, the observed behavior of users is used to organize each user into one of three categories based on likelihood of fraud (low, medium, or high). Finally, HMM is applied to these categories to detect auction fraud. Here, the authors do not provide any experimentation that would allow us to assess the effectiveness of this proposed method. Also, there is no clear bright-line to determine whether a given user is fraudulent, especially considering that the type of fraud is not specified.

In [9], Ganguly and Sadaoui propose an SB-detection model using Support Vector Machines (SVM), along with using three sampling methods (in particular, SMOTE, SpreadSubsample, and a combination of the two) in order to address the imbalance in the dataset. However, their method involves the creation of synthetic data for two missing attributes, potentially weakening results. Also, the authors do not use any kind of automated method to tune the optimal SVM parameters.

Elshaar and Sadaoui [10] use Semi-Supervised Classification (SSC) in order to identify SB in auctions. They first build on former work of Sadaoui to create a dataset with nine SB patterns. Then, they use a hybrid of data over-sampling and under-sampling in an SSC context to classify instances of SB. The authors achieve their best accuracy of 96.92% with their YATSI-J48 model. Elshaar and Sadaoui in [11] build upon their previous work cited above by implementing Cost-Sensitive Learning (CSL) within the SSC framework that they had previously established. The CSL approach aims to mitigate the impact of misclassification errors, while the SSC algorithms are trained on imbalanced data. The hybrid CSL+SSC model utilized by the authors can detect fraud with 99% accuracy and lowest cost.

Most recently, Abidi et al. [12] propose a Fussed Machine Learning approach to SB classification in real time. They use predictions from SVM and ANN modules trained in parallel on the same dataset as inputs to a fuzzy logic-based fussed module, which decides whether fraud is committed or not in each bid. If bidding behavior is abnormal, the bid is canceled and the user blocked; otherwise, no action is taken. The accuracy achieved by this approach is 99.63%.

We aim to build on the existing literature in this area by proposing a much simpler decision tree-based model which achieves a better F1 score, higher accuracy, and lower mis-classification rates than any other proposed thus far in the literature. Also, our methods determine which factors are the most predictive of SB, which no previous work has achieved.

III. DATASET

In this paper, we utilize data produced from real-world eBay auctions involving shill bidding [13]. The authors of the dataset employed the scraper Octoparse 3 to collect all the information related to completed auctions of the iPhone 7 over the period of three months, from March to June of 2017. Along with applying hierarchical clustering techniques followed by a systematic labeling technique on each cluster [14], the authors undertook a rigorous preprocessing of the scraped auction data. Their completed dataset involves information on 807 auctions and 1054 bidders, with 6321 instances; each instance is a vector consisting of an Auction ID, Bidder ID, and nine fraud predictors, which are described in depth in [14]. After this, Anowar, Sadaoui, and Mouhoub [15] used a robust two-step labeling approach to assign a class to each instance, obtaining 5646 normal instances and 675 suspicious instances. We accessed this dataset through the UCI Data Repository [16]. Of note is the large class imbalance, as there are many fewer

suspicious instances than normal instances. This imbalance creates difficulties for classification, as it encourages overclassification to the majority class. Indeed, the classification of imbalance data remains an area of active research [17].

IV. METHODOLOGY

We divide the dataset into training data and testing data (75%-25% split) using the stratified splitting method to ensure that each subset has approximately equal amounts of suspicious samples. This is important since the cardinality of the suspicious class is low. Similarly, when training we employ a five-fold cross-validation (CV) repeated three times so that each fold has enough instances of the suspicious class to avoid overclassification to the majority class.

Due to the large class imbalance, we perform automatic hyperparameter tuning by maximizing the Area Under the ROC Curve (AUC) rather than a method which optimizes accuracy, as it is most important to identify SB and prevent such users from placing bids. We do so simultaneously with the CV described above for each of three methods, all of which are based on decision trees. First, we perform Random Forests (RF) using the Ranger package; second, we perform Stochastic Gradient Boosting (GBM) using the GBM package; and third, we perform Extreme Gradient Boosting (XGB) using the XGBoost package. We use decision tree-based methods specifically because they should perform well with imbalanced data. Decision trees make splits in the feature space based on the information gain or impurity reduction criterion, and each split divides the dataset into subgroups based on a specific feature and its corresponding threshold. This localized splitting allows decision trees to identify regions of the feature space where the minority class is present, even if it is sparsely distributed.

Before we perform any of these methods for classification, we employ two different preprocessing methods in parallel to address the noted class imbalance, and we compare them to determine which is most effective. The first preprocessing method utilizes class weighting. First, we weight instances of both the majority class M and minority class m according to the formulas

$$W_M = (k)\frac{1}{|M|}$$

$$W_m = (1-k)\frac{1}{|m|}$$

where $|M|$ denotes the size of the majority class, $|m|$ the size of the minority class, and k is a hyperparameter for which we optimize. After weighting, we perform the simultaneous CV and hyperparameter tuning as previously mentioned.

The second preprocessing method involves the use of SMOTE, which involves a hybrid of over-sampling the minority class through the creation of synthetic minority class examples and randomly under-sampling the majority class, as described in [18]. Importantly, we perform SMOTE within each fold of the CV, because applying SMOTE before CV would allow synthetic samples from the minority class to appear in both the training and validation sets, resulting in overfitting. By performing SMOTE within each fold of the CV, the synthetic samples are generated independently for each fold, creating a more robust estimate of how the model will perform on unseen data. These parallel processes are illustrated below in Figure 1.

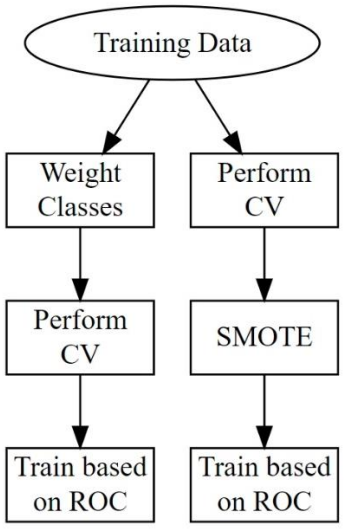

Figure 1: Parallel preprocessing of data.

The details of hyperparameter tuning for the RF method, both when we perform class weighting and when we employ SMOTE, denoted RF(W) and RF(S) respectively, can be seen below in Figure 2. The hyperparameters we tune are minimal node size, a stopping criterion that sets the minimal number of observations possible in a node of the tree, and the number of randomly selected predictors we consider at each split. The optimal combination for RF(W) is minimal node size of 1 and 3 randomly selected predictors, while the optimal combination for RF(S) is minimal node size of 1 and 9 randomly selected predictors.

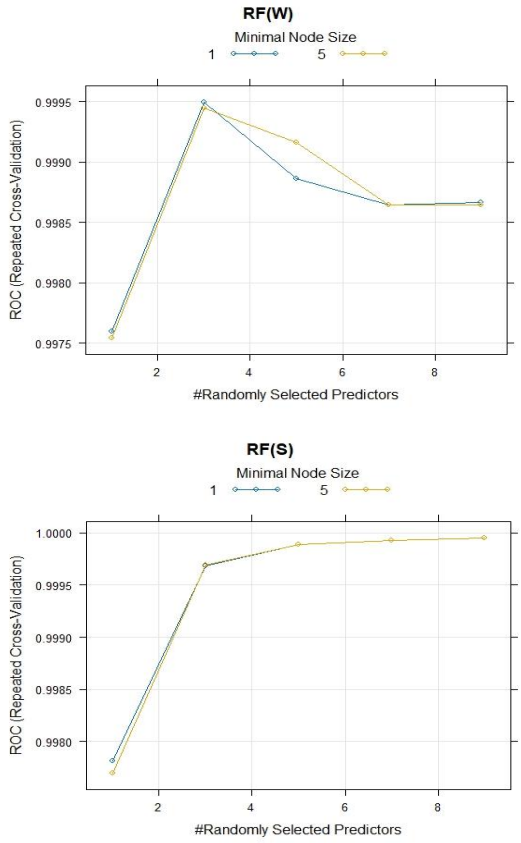

Figure 2: Hyperparameter tuning for RF models.

The hyperparameter tuning process for the GBM method is shown below in Figure 3. Unlike with RF, we can overfit by generating too many trees, so we must tune the number of boosting iterations. The shrinkage parameter determines how much the algorithm can be influenced by each new tree we produce; recall that the final model is the weighted sum of many trees. The final hyperparameter is maximum tree depth, which encodes the maximum number of splits to perform in each tree, resulting in 2N+1 terminal nodes given N splits. The optimal combination for GBM(W) is 500 boosting iterations, shrinkage parameter of 0.01, and 5 splits, while the optimal combination for GBM(S) is 100 boosting iterations, shrinkage parameter 0.2, and 5 splits.

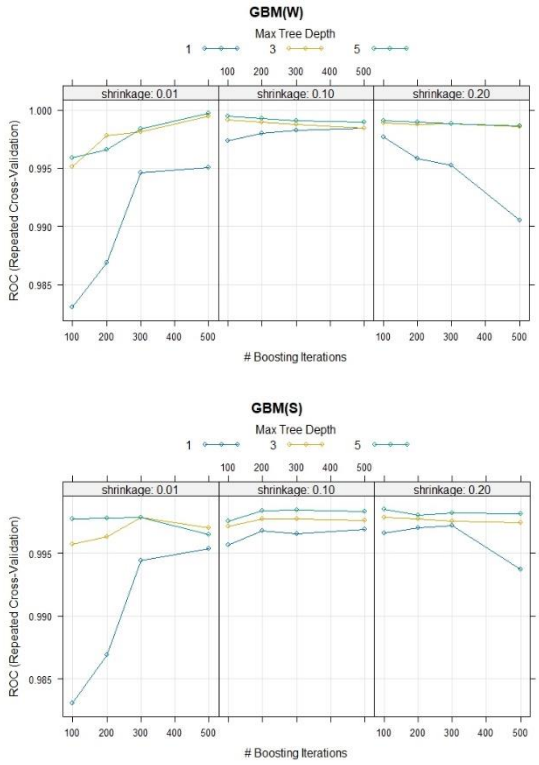

Figure 3: Hyperparameter tuning for GBM models.

Finally, the hyperparameter tuning for the XGBoost method is shown in Figure 4. We tune the same hyperparameters which we adjusted for the GBM method, and all other hyperparameters are set to their default values. The optimal combination for XGB(W) is 100 boosting iterations, shrinkage parameter of 0.1, and 6 splits, while the optimal combination for XGB(S) is 200 boosting iterations, shrinkage parameter of 0.1, and 6 splits.

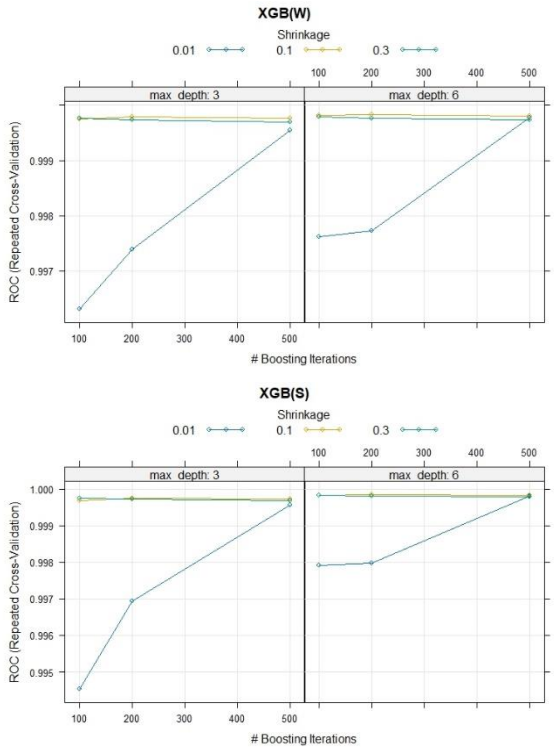

Figure 4: Hyperparameter tuning for XGB models.

V. DISCUSSION OF RESULTS

We employ various measures of misclassification in order to evaluate each of the models employed, along with the basic measure of accuracy. In terms of the number of True Positives (*TP*), i.e., the number of instances of SB which are correctly classified as SB, the number of True Negatives (*TN*), the number of False Positives (*FP*), and the number of False Negatives (*FN*), each measure of misclassification is calculated as indicated below.

$$Sensitivity = \frac{TP}{TP + FN}$$

$$Specificity = \frac{TN}{FP + TN}$$

$$PPV = \frac{TP}{TP + FP}$$

$$NPV = \frac{TN}{FN + TN}$$

$$F1\ Score = \frac{TP}{TP + \frac{1}{2}(FP + FN)}$$

We calculate each of the measures above along with the accuracy for each of the six models evaluated on the test set. These values are reported in Table I below.

TABLE I. MEASURES OF MISCLASSIFICATION

	RF(W)	*RF(S)*	*GBM(W)*	*GBM(S)*	*XGB(W)*	*XGB(S)*
Sensitivity	0.9956	0.9973	0.9894	0.9947	0.9982	0.9973
Specificity	0.9926	0.9926	0.9926	0.9926	0.9926	0.9926
PPV	0.9991	0.9991	0.9991	0.9991	0.9991	0.9991
NPV	0.9640	0.9781	0.9178	0.9571	0.9853	0.9781
Accuracy	0.9953	0.9968	0.9897	0.9945	0.9976	0.9968
F1 Score	0.9973	0.9982	0.9942	0.9967	0.9986	0.9982

Also, due to the large class imbalance in the dataset, we use F1 score as the primary measure of success. A graphical comparison of F1 scores for each model is below.

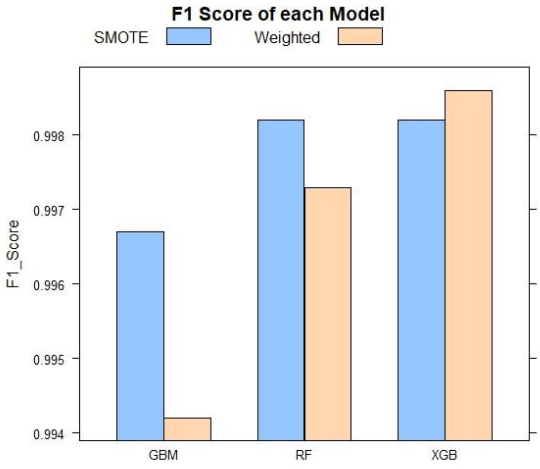

Figure 5: Bar plot of F1 Scores.

In general, it appears that preprocessing with SMOTE performs slightly better than preprocessing via class weighting. However, the best performing model overall uses class weighting and XGBoost. This model has a better F1 score and a higher accuracy than any other model

presently described in the literature related to this problem.

Finally, we look at the variable importance in each model.

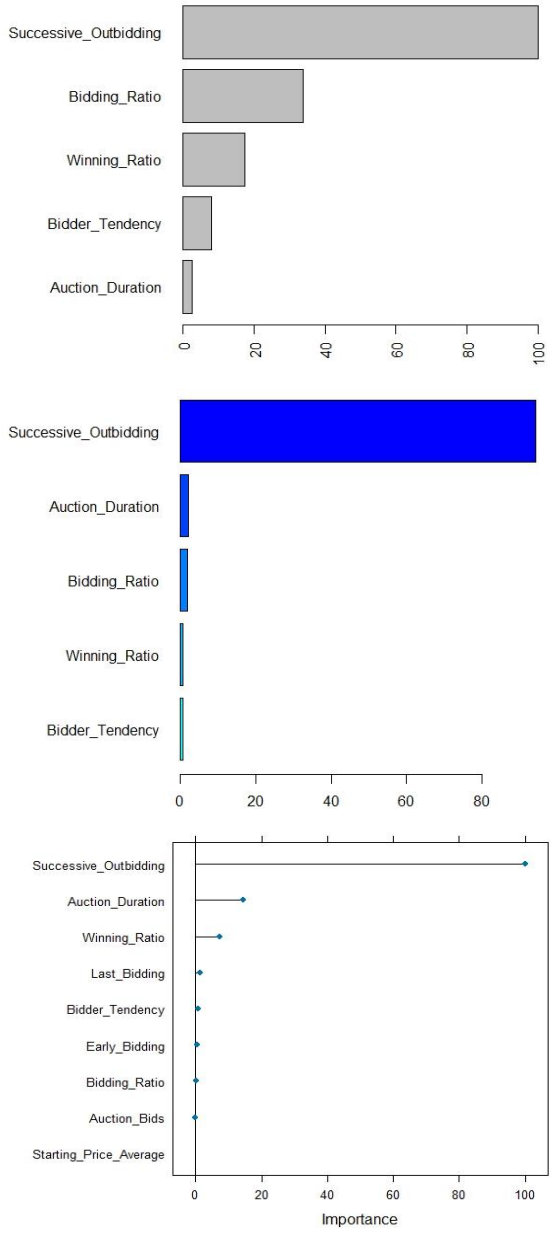

Figure 6: Variable importance plots. From top to bottom: RF, GBM, XGB.

We see that, across all models, by far the most predictive feature of SB is successive outbidding. The second most predictive feature for both GBM and XGB models is auction duration, while for RF models the second most predictive feature is bidding ratio.

VI. CONCLUSIONS AND FURTHER RESEARCH

Companies like eBay or similar hosts of online auctions should note that successive outbidding is the number one indicator of SB, so a system that tracks successive outbidding and flags users who appear to be repeatedly bidding against themselves could be effective in reducing SB.

In general, the effectiveness of all three decision tree-based methods employed here demonstrates that such methods can work even better than ANN-based methods such as those employed in [12], while having significantly lower computational cost.

It would be beneficial for future work to investigate the possibility of a decision tree-based algorithm that can be updated in real time. Work involving incremental decision trees, such as the implementation of the Hoeffding Anytime Tree described in [19], has the potential to be extremely effective in fulfilling this purpose.

ACKNOWLEDGEMENTS

The author would like to thank Dr. Randall Rojas in the Department of Economics at UCLA for invaluable instruction and support.

REFERENCES

[1] “ECommerce Sales & Size Forecast.” Trade.Gov, International Trade Administration, www.trade.gov/ecommerce-sales-size-forecast.

[2] “EBay: Number of Active Buyers 2023.” Statista, 14 Aug. 2023, www.statista.com/statistics/242235/number-of-ebays-total-active-users/.

[3] Mamata Jenamani, Yuhui Zhong, Bharat Bhargava, “Cheating in online auction – Towards explaining the popularity of English auction,” Electronic Commerce Research and Applications, Volume 6, Issue 1, 2007,

Pages 53-62, ISSN 1567-4223, https://doi.org/10.1016/j.elerap.2005.12.002.
[4] Majadi, Nazia, Trevathan, Jarrod, & Gray, Heather. (2018). A Run-Time Algorithm for Detecting Shill Bidding in Online Auctions. Journal of Theoretical and Applied Electronic Commerce Research, 13(3), 17-49. https://dx.doi.org/10.4067/S0718-18762018000300103.
[5] J. Trevathan and W. Read, "Undesirable and fraudulent behaviour in online auctions," in Proceedings of International Conference on Security and Cryptography, Portugal, 2006, pp. 450-458.
[6] F. Dong, S. M. Shatz, and H. Xu, "Combating online in-auction fraud: Clues, techniques and challenges," Computer Science Review, vol. 3, no. 4, pp. 245–258, 2009.
[7] Benjamin J. Ford, Haiping Xu, Iren Valova, A Real-Time Self-Adaptive Classifier for Identifying Suspicious Bidders in Online Auctions, The Computer Journal, Volume 56, Issue 5, May 2013, Pages 646–663, https://doi.org/10.1093/comjnl/bxs025
[8] P. Gupta and A. Mundra, "Online in-auction fraud detection using online hybrid model," International Conference on Computing, Communication & Automation, Greater Noida, India, 2015, pp. 901-907, doi: 10.1109/CCAA.2015.7148504.
[9] Ganguly, S., Sadaoui, S. (2018). "Online Detection of Shill Bidding Fraud Based on Machine Learning Techniques." In: Mouhoub, M., Sadaoui, S., Ait Mohamed, O., Ali, M. (eds) Recent Trends and Future Technology in Applied Intelligence. IEA/AIE 2018. Lecture Notes in Computer Science, vol 10868. Springer, Cham. https://doi.org/10.1007/978-3-319-92058-0_29
[10] S. Elshaar and S. Sadaoui, "Semi-supervised classification of fraud data in commercial auctions," Appl. Artif. Intell., vol. 34, no. 1, pp. 47–63, Jan. 2020, doi: 10.1080/08839514.2019.1691341.
[11] S. Elshaar and S. Sadaoui, "Cost-sensitive semi-supervised classification for fraud applications," in Proc. 12th Int. Conf. (ICAART). Valletta, Malta: Springer, 2020, Feb. 2020, pp. 173–187.
[12] Abidi, Wajhe Ul Husnain et al. (2021). Real-Time Shill Bidding Fraud Detection Empowered With Fussed Machine Learning. IEEE Access, 9, 113612-113621.
[13] Alzahrani, Ahmad & Sadaoui, Samira. (2018). Scraping and Preprocessing Commercial Auction Data for Fraud Classification. https://doi.org/10.48550/arXiv.1806.00656
[14] Alzahrani, Ahmad & Sadaoui, Samira. (2018). Clustering and Labelling Auction Fraud Data. https://doi.org/10.48550/arXiv.1808.07288
[15] F. Anowar, S. Sadaoui and M. Mouhoub, "Auction Fraud Classification Based on Clustering and Sampling Techniques," 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA), Orlando, FL, USA, 2018, pp. 366-371, doi: 10.1109/ICMLA.2018.00061.
[16] Shill Bidding Dataset. (2020). UCI Machine Learning Repository. https://doi.org/10.24432/C5Z611.
[17] S. Zhang, S. Sadaoui, and M. Mouhoub, "An empirical analysis of imbalanced data classification," Computer and Information Science, vol. 8, no. 1, pp. 151–162, 2015.
[18] Chawla, Nitesh V., et al. "SMOTE: synthetic minority over-sampling technique." Journal of artificial intelligence research 16 (2002): 321-357.
[19] Manapragada, Chaitanya, Geoffrey I. Webb, and Mahsa Salehi. "Extremely fast decision tree." *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2018.