A Comparative Analysis of the Ensemble Models for Detecting GPS Spoofing attacks on UAVs

Aydan Gasimova, Tala Talaei Khoei, and Naima Kaabouch
School of Electrical Engineering and Computer Science, University of North Dakota
Grand Forks, ND 58202 USA

2022 IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC) | 978-1-6654-8303-2/22/$31.00 ©2022 IEEE | DOI: 10.1109/CCWC54503.2022.9720738

***Abstract*—Unmanned Aerial Vehicles have been widely used in military and civilian areas. The positioning and return-to-home tasks of UAVs deliberately depend on Global Positioning Systems (GPS). However, the civilian GPS signals are not encrypted, which can motivate numerous cyber-attacks on UAVs, including Global Positioning System spoofing attacks. In these spoofing attacks, a malicious user transmits counterfeit GPS signals. Numerous studies have proposed techniques to detect these attacks. However, these techniques have some limitations, including low probability of detection, high probability of misdetection, and high probability of false alarm. In this paper, we investigate and compare the performances of three ensemble-based machine learning techniques, namely bagging, stacking, and boosting, in detecting GPS attacks. The evaluation metrics are the accuracy, probability of detection, probability of misdetection, probability of false alarm, memory size, processing time, and prediction time per sample. The results show that the stacking model has the best performance compared to the two other ensemble models in terms of all the considered evaluation metrics.**

Keywords*—Unmanned Aerial Vehicles, GPS spoofing attacks, spoofing detection, machine learning, ensemble models, cyber-attacks, classification, cybersecurity.

I. Introduction

There has been an increased interest in Unmanned Aerial Vehicles (UAVs) for civil applications over the last decade [1]. Several tasks of UAVs, including navigation, positioning, and return-to-home, are dependent on Global Positioning System (GPS) devices. The civilian GPS signals are not encrypted, and as a result, can be easily spoofed which is a problem for UAV's safe flight operations [1].

For this purpose, a number of techniques have been proposed to detect and mitigate such attacks. Some of these techniques are hardware-based [2 - 4]. For example, in [2], the authors proposed a method using the off-the-shelf global navigation satellite system (GNSS) antennas to detect GPS spoofing attacks. This method is able to detect malicious signals from diverse locations. In [3], the authors presented a GPS spoofing detection algorithm using the Doppler frequency difference of arrival in a dual-antenna receiver, that exploits the regularity between the signal features and the navigational information. In [4], the authors proposed a detection technique and removal method which relies on the existing Cooperative Adaptive Cruise Control system to provide inter-vehicle ranging and data sharing.

Other studies proposed GPS spoofing attack detection techniques using artificial intelligence (AI) methods, including machine learning (ML) [5-8]. For instance, in [5], the authors presented an ML technique based on Support Vector Machines in detection of malicious signals on UAVs. In [6], the authors proposed an anomaly detection technique based on K-nearest Neighbors (KNN) to detect GPS spoofing on UAVs. In [7], the authors provided a comparison performance of tree-based Supervised machine learning models for detecting GPS spoofing attacks on UAVs. In [8], the authors compared the performance of two ML models, namely KNN and Naive Bayesian (NB) classifiers, to detect attacks on UAVs.

All these hardware and AI-based techniques show good accuracy. However, a performance comparison should be made in terms of different performance metrics, such as probabilities of misdetection and false alarm, which are missing from several studies. In addition, most of the works did not employ hyperparameter tuning techniques to improve the performance of the algorithms. This study fills the existing gap by enhancing the performance of ML models using ensemble-based ML models with a particular hyperparameter technique. These ensemble-learning techniques are bagging, stacking, and boosting. Each of these models incorporates the decisions of several machine learning models to increase the detection performance. To identify the correlated features in the dataset, we choose Pearson's Correlation Coefficient. In addition, to optimize the results, we use the hyperparameter tuning technique, Grid search, to find the best hyperparameters for each model. These ensemble models are compared based on the evaluation metrics: accuracy, probability of detection, probability of misdetection, probability of false alarm, memory size, processing time, and prediction time per sample.

The following summarizes the contribution of this study:

- Identification of the most important features by using two feature selection methods.
- Comparative analysis of ensemble learning methods using seven evaluation metrics, namely accuracy, probability of detection, probability of misdetection, probability of false alarm, memory size, processing time, and pre-

978-1-6654-8303-2/22/$31.00 ©2022 IEEE

diction time per sample.

The remainder of this paper is organized as follows: Section II outlines the materials and the methodology used in this study. The results are described in Section III. Finally, the conclusion and future work are drawn in Section IV.

II. Materials and methods

This section describes the dataset collection, features, and data preprocessing techniques, including data cleaning and normalization. We also briefly describe the feature selection methods and classification models.

A. Data description

To collect GPS signals, a software-defined radio receiver was employed to collect GPS signals at different speeds, positions, and altitudes [7]. Three types of spoofing attacks were simulated, namely simplistic, intermediate, and sophisticated. In simplistic spoofing attacks, the broadcasted signals are not synchronized with the authentic GPS signals; therefore, these attacks can be easily detected. In intermediate spoofing, attack estimate the target receiver antenna location and velocity before broadcasting the fake GPS signals. Sophisticated spoofing attacks are the most advanced type of GPS spoofing attacks, in which multiple synchronized phase-locked intermediate spoofers are used to avoid the detection by the target receiver [7].

The dataset for training and testing is built by identifying 13 features. These features, along with their equations, and descriptions are discussed in detail in Table I.

B. Data preprocessing

In this work, we used a dataset consisting of 10,056 samples. This dataset includes 4,764 attack samples and 5,382 authentic samples. Data corresponding to GPS spoofing attacks are encoded as 1, and the remaining are encoded as 0. The final step is to normalize data into the compatible form for modeling. Several techniques exist to normalize raw data, including Mean and Standard Deviation Based Normalization Methods, Decimal Scaling Normalization, and Median and Median Absolute Deviation Normalization. In this study, we used the min-max scalar to normalize data. The min-max normalization inserts the data into a common scale, which increases the performance of the classifiers [8, 9]. Each feature's value is scaled to a number between 0 and 1. The min-max scalar is calculated as [9]:

$$X = \frac{x - Min(x)}{Max(x) - Min(x)} \tag{1}$$

where *x* is the initial value, *Min(x)* and *Max(x)* are the minimum and maximum values of the feature vector.

C. Feature selection

Feature selection is a crucial step to identify the significant features to obtain high-performance results. Correlated or irrelevant features in the dataset can affect the performance of the classification models. In this study, we employed a correlation technique, Pearson Correlation, which is a filter-based method that measures the correlation between two variables and ascertain the strength of the linear connection ranging from -1 to 1. When the result is close to 1 or -1, the characteristics have a strong connection, either positive or negative. A positive correlation coefficient indicates positive linear correlations, while a negative correlation coefficient indicates negative linear correlations [5]. The correlation coefficient is given by:

$$R_p = \frac{\sum_{i=1}^{n}[(x_i - \bar{x})(y_i - \bar{y})]}{\sqrt{\sum_{i=1}^{n}[(x_i - \bar{x})^2]}\sqrt{\sum_{i=1}^{n}[(y_i - \bar{y})^2]}} \tag{2}$$

where n is a sample size, *x* and *y* are two variables, $\bar{x}$ and $\bar{y}$ are the means of the two variables, and x_i and y_i are the individual sample points indexed with *i*. In general, a weak correlation corresponds to a value of R_p less than or equal to 0.39 and a moderate correlation is defined as an R_p value between 0.40 and 0.89. In this study, we considered a correlation coefficient threshold of 0.88 and -0.88 for highly correlated features.

In addition to feature selection methods, a static relationship is needed for a proper model since ML algorithms cannot handle non-stationary data modification. We search for characteristics that follow a non-stationary distribution and use analysis to transform the raw data to stationary data, includes determining the consecutive deviations between samples. This method is equated below [7]:

$$R = \frac{x_{i+1} - x_i}{n_{i+1} - n_i} \tag{3}$$

where *R* is the rate of change and $n_{i+1} - n_i$ is the distance between two instances, which is equal to 1.

D. Classification models

Traditional machine learning techniques may not always produce high performance results, particularly when the data is composite or unbalanced [10]. One possible way is to use an ensemble learning model, which is a set of training models that work together to enhance the accuracy of a single model's predictions [11 - 13]. In general, these techniques can be classified into three categories, namely bagging, stacking, and boosting models. The stacking technique employs several classification algorithms by using their outputs as inputs of a final estimator to obtain high accuracy. This technique is executed at two levels. In first level, the algorithm mainly trains various models including their prediction results, while in second level the model evaluates the best estimate of previous level predictions [11]. In the bagging model, multiple

TABLE I: LIST OF EXTRACTED FEATURES.

Extracted features.	Abbreviations	Descriptions
Carrier to Noise Ratio	C/N0	Indicator of the signal that carries the GPS information.
Magnitude of the Early Correlator	EC	Magnitudes of the Early correlator are used for timing recovery.
Magnitude of the Late Correlator	LC	Magnitudes of the Late correlator are used for timing recovery.
Magnitude of the Prompt Correlator	PC	Estimation of phase and frequency differences.
Prompt in-phase correlator	PIP	In-phase signal of the prompt correlator.
Prompt Quadrature component	PQP	Quadrature signal of the prompt correlator.
Carrier Doppler in Tracking loop	TCD	Carrier Loop Doppler Measurements.
Carrier Doppler	DO	Change in frequency for a GPS receiver.
Pseudo-range	PD	Time difference between transmission and reception time.
Receiver Time	RX	Time of reception after the start of time of the week.
Time of the week	TOW	Time of the transmission of the navigation message.
Carrier Phase Cycles	CP	Frequency difference between the received carrier and a receiver-generated carrier phase.
Satellite vehicle number	PRN	Identification of different satellites orbiting the earth.

evaluations are calculated, and the average of them is used to make the prediction. This model has several ML estimators that use decision trees and individual learners to make a prediction [12]. One advantage of such a method is to reduce the base algorithm's choice and increase the accuracy of the model. In the boosting model, weak learners are converted into strong ones by collecting algorithms. At each iteration, the learning of this model is done according to the training weights, which is updated based on the previous iteration's performance. To improve the classification results, the boosting model uses a technique known as decision trees that combines several models with varying levels of performance [13].

In this study, we combine five different traditional classification methods, namely KNN, NB, decision tree (DT), random forest, and logistic regression, in the stacking technique. In the bagging method, to achieve the best results, we apply DT classification technique. Finally, we implement the Gradient Tree Boost ensemble model in the boosting model.

III. RESULTS

To train and test the proposed algorithms, we employ a 10-fold cross-validation method. With this technique, the models are trained with 80% and tested with 20% of data in the given dataset. A comparison between these algorithms is carried out based on several evaluation metrics: the accuracy, probability of detection P_d, the probability of misdetection P_{md}, and the probability of false alarm P_{fa}. We calculate these metrics by applying the following equations:

$$Accuracy = \frac{(T_P + T_N)}{(T_P + T_N + F_P + F_N)} \tag{4}$$

$$P_d = \frac{T_P}{(T_P + F_N)} \tag{5}$$

$$P_{md} = \frac{F_N}{(T_P + F_N)} \tag{6}$$

$$P_{fa} = \frac{F_P}{(F_P + T_N)} \tag{7}$$

where T_P is the number of correct predicted malicious attacks, T_N is the number of correct predicted normal attacks, F_P is the number of incorrect predicted malicious attacks, and F_N is the number of incorrect normal attacks.

Due to the size, weight, and power (SWaP) constraints, subsidiary three metrics are employed: memory size of each model, the processing time to perform all steps, and the average prediction time per instance. The memory size observes the consumption of the memory for each model separately, as well as line-by-line investigation of memory use. The processing time refers to the prerequisite time to train and test the models, and it highly depends on the used ML classifier. The average prediction time for each instance is prerequisite to predict whether the current sample is authentic.

The results of the investigation are presented in Figs.1 to 2 and Tables II to IV.

TABLE II: PARAMETERS FOR CLASSIFICATION MODELS.

Model	Best parameters
Stacking	n_estimators = 42.
Bagging	final_estimator_verbose= 1.
Boosting	max_depth= 10, min_impurity_decrease = 10.

After implementing the models and metrics, we applied the Grid search as a hyperparameter tuning technique to obtain the best results for each model. These hyperparameters are described in Table II for each of the three ensemble models.

Fig. 1. illustrates the results of Pearson's correlation coefficient for each pair of the features. As one can observe, few features are highly correlated. We selected the threshold 0.9 to identify highly correlated features. As a result of this method, RX and DO, are considered highly correlated with TOW, and TCD, respectively.

Due to their lower importance than that of DO and TOW, RX and TCD were discarded from the list of features. Finally, eleven features, namely DO, TOW, PD, CP, CN0, PRN, PC, PQP, PIP, LC, and EC are considered relevant and uncorrelated features for classifying GPS spoofing attacks on UAVs. Moreover, subsidiary data preprocessing step is

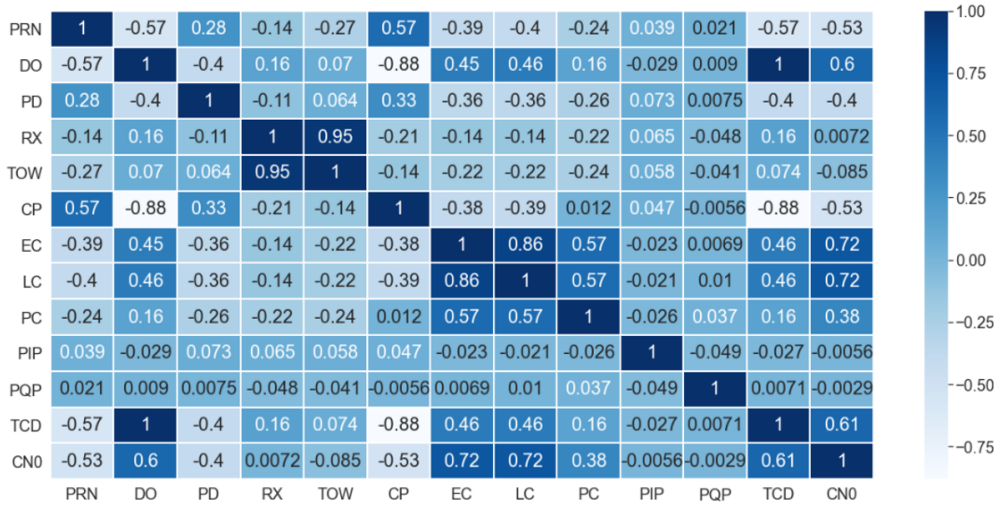

Fig. 1: Pearson's Correlation Coefficient Heatmap.

implemented to transform two features, namely TOW and CP, into stationary distribution.

The results of the three selected algorithms are shown in Fig. 2 in terms of accuracy, probability of detection, probability of misdetection, and probability of false alarm. Fig. 2a shows the accuracy of the bagging, stacking, and boosting models. As one can see, the stacking model has the highest accuracy (95.43%), followed by the bagging (95.28%), then the boosting model (94.61%). Therefore, these results show that the stacking model provides the best accuracy for detecting GPS spoofing attacks. However, the accuracy is not sufficient to compare the efficiency of ML models in detecting GPS spoofing attacks. The number of falsely detected alarms, and misdetected samples can degrade the performance of ML models.

Fig. 2b shows the results of the selected models in terms of the probability of detection. As one can see, the stacking classifier has the highest detection probability of 99.56%, the bagging classifier has a detection probability of 99.24%, and the boosting model has a detection probability of 96.55%, which is considered the lowest result compared to the two other ensemble models.

Fig. 2c shows the probability of misdetection of the selected ensemble models. As one can see, the stacking classifier has a probability of misdetection of 0.36%, the bagging model shows a probability of misdetection of 0.64%, and the boosting model has a probability of misdetection of 2.95%. Consequently, the stacking model has the lowest probability of misdetection, whereas the boosting model has the highest and worse probability of misdetection.

Fig. 2d illustrates the results of the probability of false alarm of the selected models. As one can see, the stacking classifier has the best result in terms of the probability of false alarm (0.43%), followed by the bagging model (1.07%) and then boosting classifier (5.08%). The summary of the performance results of the proposed models in terms of the four evaluation metrics are given in Table III.

TABLE III: EVALUATION METRICS.

Classifiers	**Accuracy (%)**	**P_d (%)**	**P_{md} (%)**	**P_{fa} (%)**
Bagging	95.28	99.24	0.64	1.07
Stacking	95.43	99.56	0.36	0.03
Boosting	94.61	96.55	2.95	5.08

TABLE IV: SIZE AND METRICS OF COMPARED CLASSIFIERS.

ML Classifier	**Model size in memory (Mb)**	**Processing time (Sec)**	**Average prediction time per sample (Sec)**
Bagging	190.4	0.74	0.02
Stacking	191.3	13.06	0.24
Boosting	190.5	1.5	0.01

Table IV gives the results of the memory size of each model, the processing time, and average prediction time of each sample for each model. As one can see in this table, the stacking classifier presents the worst outcomes in terms of processing time and average prediction time compared with the other ensemble techniques. In addition, the stacking classifier employs the biggest proportion of memory size (191.3 megabytes), followed by the bagging model (190.4 megabytes), then the boosting method (190.5 megabytes). The stacking model has a processing time of 13.06 seconds, the bagging model has 0.74 seconds, and boosting model has 1.5 seconds. As a result, the bagging classifier provides the best results in terms of processing time, followed by bagging and stacking models. Finally, the stacking classifier has the worst average prediction time of 0.24 seconds per instance, followed

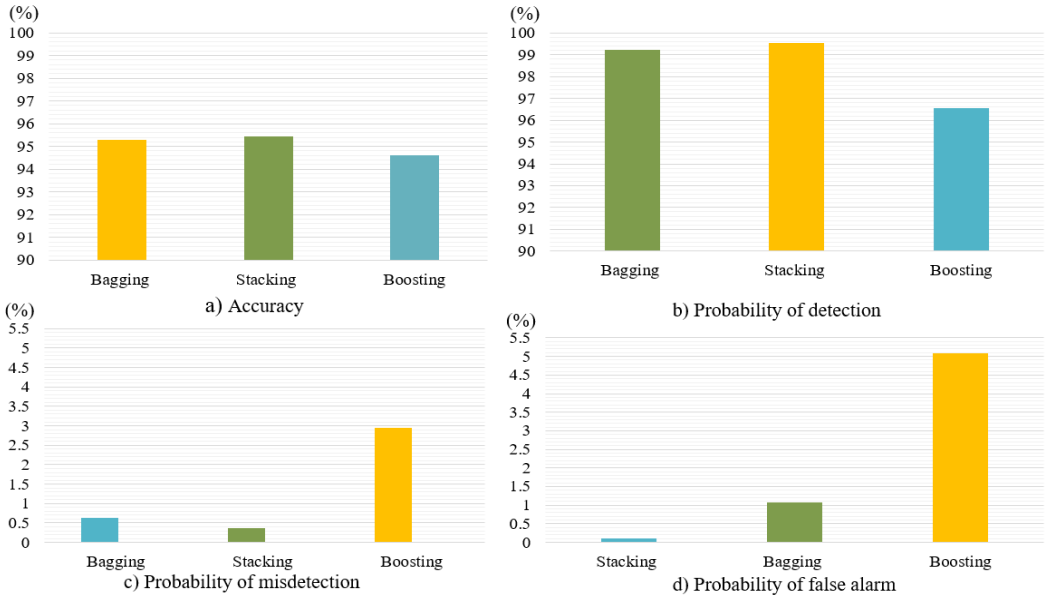

Fig. 2: Evaluation Results for the GPS Spoofing Attacks.

by the bagging with a prediction time of 0.02 seconds and the boosting model with 0.01 seconds.

In summary:

- The correlated features, namely RX and TCD are discarded from the corresponding dataset.
- The Grid search hyperparameter tuning method is used to find the best hyperparameters for each model.
- Among the ensemble models, the stacking model gives the best results in terms of probabilities of detection, misdetection, and false alarm. Whereas it has the highest processing time and average prediction time for each instance.
- The boosting classifier provides the lowest results compared to the bagging and stacking models.
- The bagging model has good detection, misdetection, and false alarm probabilities that are slightly below those of the stacking algorithm. However, its memory size, processing time and average prediction time per sample are 1, 18, and 12 times smaller than those of the stacking algorithm.

IV. Conclusion

GPS spoofing attacks are among the most important threats that target UAVs. In this paper, we presented a performance comparison of the bagging, stacking, and boosting algorithms in detecting GPS spoofing attacks in terms of accuracy, probability of detection, probability of misdetection, probability of false alarm, memory size, processing time, and prediction time per sample. First, we identified the most relevant and uncorrelated features using Pearson's Correlation feature selection technique. The results show that RX and TCD are highly correlated and have lower importance scores, thus they are discarded from the corresponding dataset. In addition, we implemented the Grid search technique for hyperparameter tuning to determine the optimal hyperparameters for each model. The simulation results show that the stacking-based ensemble learning model has the best results compared to the bagging and boosting classifiers. In contrast, the boosting classifier provides the lowest results among all ensemble models. For future work, we plan to investigate the performance of deep learning models in detecting GPS spoofing attacks on UAVs.

Acknowledgement

The authors acknowledge the support of the National Science Foundation (NSF), Award Number 2006674.

References

[1] M. R. Manesh and N. Kaabouch, "Cyber-attacks on unmanned aerial system networks: Detection, countermeasure, and future research directions," *Computers & Security*, vol. 85, pp. 386–401, 2019.

[2] J. Chen, Y. Xu, H. Yuan, and Y. Yuan, "A new GNSS spoofing detection method using two antennas," *IEEE Access*, vol. 8, pp. 110 738–110 747, 2020.

[3] L. He, H. Li, and M. Lu, "Dual-antenna GNSS spoofing detection method based on doppler frequency difference of arrival," *GPS Solutions*, vol. 23, no. 3, pp. 1–14, 2019.

[4] N. Carson, S. M. Martin, J. Starling, and D. M. Bevly, "GPS spoofing detection and mitigation using cooperative adaptive cruise control system," in *IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2016, pp. 1091–1096.

[5] S. Semanjski, A. Muls, I. Semanjski, and W. De Wilde, "Use and validation of supervised machine learning approach for detection of GNSS signal spoofing," in *International Conference on Localization and GNSS (ICL-GNSS)*. IEEE, 2019, pp. 1–6.

[6] G. Liu, R. Zhang, C. Wang, and L. Liu, "Synchronization-free GPS spoofing detection with crowdsourced air traffic control data," in *IEEE International Conference on Mobile Data Management (MDM)*. IEEE, 2019, pp. 260–268.
[7] G. Aissou, H. Ould Slimane, S. Benouadah, and N. Kaabouch, "Tree-based supervised machine learning models for detecting GPS spoofing attacks on UAS," in *IEEE Ubiquitous Computing, Electronics Mobile Communication Conference (UEMCON)*. IEEE, 2021, pp. 0649–0653.
[8] E. Shafiee, M. Mosavi, and M. Moazedi, "Detection of spoofing attack using machine learning based on multi-layer neural network in single-frequency GPS receivers," *The Journal of Navigation*, vol. 71, no. 1, pp. 169–188, 2018.
[9] D. Singh and B. Singh, "Investigating the impact of data normalization on classification performance," *Applied Soft Computing*, vol. 97, p. 105524, 2020.
[10] M. H. D. M. Ribeiro and L. dos Santos Coelho, "Ensemble approach based on bagging, boosting and stacking for short-term prediction in agribusiness time series," *Applied Soft Computing*, vol. 86, p. 105837, 2020.
[11] T. T. Khoei, G. Aissou, W. C. Hu, and N. Kaabouch, "Ensemble learning methods for anomaly intrusion detection system in smart grid," in *2021 IEEE International Conference on Electro Information Technology (EIT)*. IEEE, 2021, pp. 129–135.
[12] J. Dou, A. P. Yunus, D. T. Bui, A. Merghadi, M. Sahana, Z. Zhu, C.-W. Chen, Z. Han, and B. T. Pham, "Improved landslide assessment using support vector machine with bagging, boosting, and stacking ensemble machine learning framework in a mountainous watershed," *Landslides*, vol. 17, no. 3, pp. 641–658, 2020.
[13] T. Talaei Khoei, S. Ismail, and N. Kaabouch, "Dynamic selection techniques for detecting GPS spoofing attacks on UAVs," *Sensors*, vol. 22, no. 2, 2022. [Online]. Available: https://www.mdpi.com/1424-8220/22/2/662