LAB 3 Report

Course Name: Machine Learning

Course Code: CSE-475

Section - 3

Lab Name: Ensemble Learning and Explainable AI

Submitted By
Name: Ismail Mahmud Nur
ID: 2021-2-60-052
Dept. of Computer Science & Engineering

Submitted To
Dr Raihan Ul Islam,
Associate Professor,
Department of Computer Science and Engineering
East West University

Cyber Threat Detection with Ensemble Learning and Explainable AI

Abstract

The goal of this study is to explore and evaluate various ensemble learning techniques for cyber threat detection using a labeled dataset. We apply four main ensemble methods—Bagging (Random Forest), Boosting (AdaBoost and Gradient Boosting), Stacking, and Voting Classifiers—on a real-world dataset. The performance of these models is evaluated using accuracy, precision, recall, and F1-score. Besides, we use Explainable AI (XAI) methods to provide interpretation of the model predictions and understand the contribution of various features to the classification decisions. Our results suggest that ensemble methods, in particular Stacking, perform very well on threat detection tasks, while SHAP and LIME make models more interpretable.

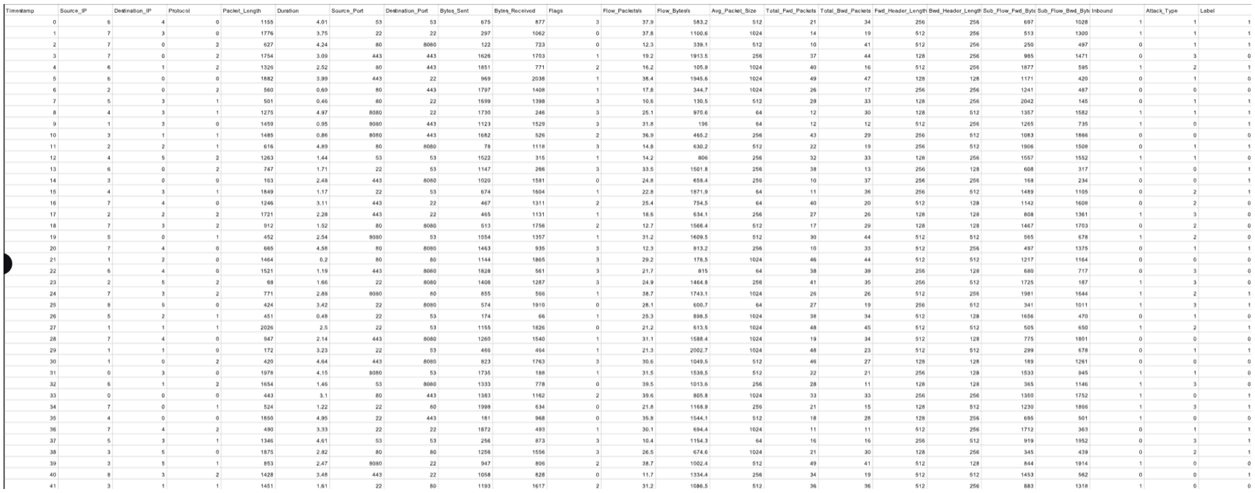
1. Introduction

Cybersecurity threats are always evolving, and early detection of such threats is very important to protect the systems and sensitive data. Traditional machine learning models have been widely applied to classification tasks like identifying malicious network activity or predicting possible security breaches. In this report, we take up the application of ensemble learning techniques for cyber threat detection and compare models such as Random Forest, AdaBoost, Gradient Boosting, Stacking, and Voting Classifiers. Ensemble techniques are known to increase the effectiveness of single models by combining their predictions. Besides, with the rise of Explainable Artificial Intelligence (XAI), it is our aim to gain insight into the decision-making process of such models using SHAP and LIME, which help in interpreting complex machine learning models.

2. Methodology

2.1 Dataset

The dataset used in the study is that of cyber threat detection data, where each instance represents an individual network activity and a target variable indicating "Normal" or "Malicious," based on the nature of activity. It includes several network activity description features, such as packet size, protocol, duration, among others. Below is an analysis of preprocessing to remove any unnecessary feature and handling missing values.



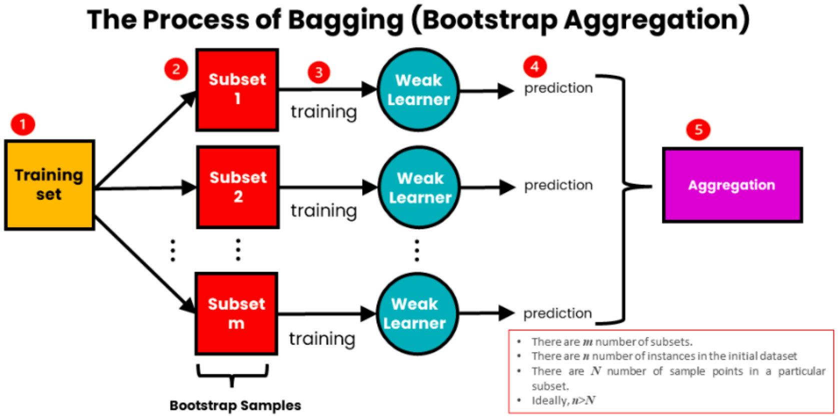
Timestamp	Source_IP	Destination_IP	Protocol	Packet_Length	Duration	Source_Port	Destination_Port	Bytes_Sent	Bytes_Received	Flags	Flow_Packets/s	Flow_Bytes/s	Avg_Packet_Size	Total_Fwd_Packets	Total_Bwd_Packets	Fwd_Header_Length	Bwd_Header_Length	Sub_Flow_Fwd_Byte	Sub_Flow_Bwd_Byte	Inbound	Attack_Type	Label
0	6	4	0	1155	4.01	53	53	675	877	3	37.9	583.2	512	21	34	256	256	697	1028	1	1	1
1	7	3	0	1776	3.75	22	22	297	1062	0	37.8	1100.6	1024	14	19	512	256	513	1300	1	1	1
2	7	0	2	627	4.24	80	8080	122	723	0	12.3	339.1	512	10	41	512	256	250	497	0	1	1
3	7	0	2	1754	3.09	443	443	1626	1703	1	19.2	1913.5	256	37	44	128	256	985	1471	0	3	0
4	6	1	2	1326	2.52	80	443	1851	771	2	16.2	105.9	1024	40	16	512	256	1877	595	1	2	1
5	6	0	0	1882	3.99	443	22	969	2038	1	38.4	1945.6	1024	49	47	128	128	1171	420	0	1	0
6	2	0	2	560	0.69	80	443	1797	1408	1	17.8	344.7	1024	26	17	256	256	1241	487	0	0	0
7	5	3	1	501	0.46	80	22	1699	1398	3	10.6	130.5	512	29	33	128	256	2042	145	0	1	1
8	4	3	1	1275	4.97	8080	22	1730	246	3	25.1	970.6	64	12	30	128	512	1357	1582	1	1	1
9	1	3	0	1459	0.95	8080	443	1123	1529	3	31.8	196	64	12	12	512	256	1265	735	1	0	1
10	3	1	1	1485	0.86	8080	443	1682	526	2	36.9	465.2	256	43	29	256	512	1083	1866	0	0	0
11	2	2	1	616	4.89	80	8080	78	1118	3	14.8	630.2	512	22	19	256	512	1906	1508	0	1	1
12	4	5	2	1263	1.44	53	53	1522	315	1	14.2	806	256	32	33	128	256	1557	1552	1	1	0
13	6	0	2	747	1.71	22	53	1147	266	3	33.5	1501.8	256	38	13	256	128	608	317	1	0	1
14	3	0	0	163	2.48	443	8080	1020	1581	0	24.8	658.4	256	10	37	256	256	168	234	0	0	1
15	4	3	1	1849	1.17	22	53	674	1604	1	22.8	1871.9	64	11	36	256	512	1489	1105	0	2	1
16	7	4	0	1246	3.11	443	22	467	1311	2	25.4	754.5	64	40	20	512	128	1142	1608	0	2	0
17	2	2	2	1721	2.28	443	22	465	1131	1	18.6	634.1	256	27	26	128	128	808	1361	1	3	0
18	7	3	2	912	1.52	80	8080	513	1756	2	12.7	1566.4	512	17	29	128	128	1467	1703	0	2	0
19	5	0	1	452	2.54	8080	53	1554	1357	1	31.2	1609.5	512	30	44	512	512	565	678	1	2	0
20	7	4	0	665	4.58	80	8080	1463	935	3	12.3	813.2	256	10	33	512	256	497	1375	0	1	1
21	1	2	0	1464	0.2	80	80	1144	1865	3	29.2	178.5	1024	46	44	512	512	1217	1164	0	0	0
22	6	4	0	1521	1.19	443	8080	1828	561	3	21.7	815	64	38	39	256	128	680	717	0	3	0
23	2	5	2	68	1.66	22	8080	1406	1287	3	24.9	1464.8	256	41	35	256	512	1725	187	1	3	0
24	7	3	2	771	2.88	8080	80	855	566	1	38.7	1743.1	1024	26	26	512	256	1981	1644	1	2	1
25	8	5	0	424	3.42	22	8080	574	1910	0	28.1	600.7	64	27	19	256	512	341	1011	1	3	1
26	5	2	1	451	0.48	22	53	174	66	1	25.3	898.5	1024	38	34	512	128	1656	470	0	1	0
27	1	1	1	2026	2.5	22	53	1155	1826	0	21.2	613.5	1024	48	45	512	512	505	650	1	2	1
28	7	4	0	947	2.14	443	8080	1260	1540	1	31.1	1588.4	1024	19	34	512	128	775	1801	0	0	0
29	1	1	0	172	3.23	22	53	466	464	1	21.3	2002.7	1024	48	23	512	512	299	678	0	1	1
30	1	0	2	420	4.64	443	8080	823	1763	3	30.6	1049.5	512	46	27	128	128	189	1261	0	0	0
31	0	3	0	1978	4.15	8080	53	1735	188	1	31.5	1539.5	512	22	21	256	128	1533	945	1	1	0
32	6	1	2	1654	1.46	53	8080	1333	778	0	39.5	1013.6	256	28	11	128	128	365	1146	1	3	0
33	0	0	0	443	3.1	80	443	1383	1162	2	39.6	805.8	1024	33	33	256	256	1350	1752	1	0	1
34	7	0	1	524	1.22	22	80	1998	634	0	21.8	1168.9	256	21	15	128	512	1230	1866	1	3	1
35	4	0	0	1850	4.95	22	443	181	968	0	35.8	1544.1	512	18	28	128	256	695	501	1	0	0
36	7	4	2	490	3.33	22	22	1872	493	1	30.1	694.4	1024	11	11	512	256	1712	363	0	1	1
37	5	3	1	1346	4.61	53	53	256	873	3	10.4	1154.3	64	16	16	256	512	919	1952	0	3	1
38	3	5	0	1875	2.82	80	80	1256	1556	3	26.5	674.6	1024	21	30	128	256	345	439	0	2	1
39	3	5	1	853	2.47	8080	22	947	806	2	38.7	1002.4	512	49	41	512	128	844	1914	1	0	0
40	8	3	2	1428	3.48	443	22	1058	828	0	11.7	1334.4	256	34	19	512	512	1453	562	0	0	1
41	3	1	1	1451	1.61	22	80	1193	1617	2	31.2	1086.5	512	36	36	512	256	883	1318	1	0	0

Figure 2.1 : Dataset Preview

2.2 Ensemble Learning Techniques

We applied four primary ensemble techniques to the dataset, as described below:

Bagging (Random Forest): Random Forest, a Bagging method, creates an ensemble of decision trees to reduce variance and improve the model's performance.



- There are m number of subsets.
- There are n number of instances in the initial dataset
- There are N number of sample points in a particular subset.
- Ideally, $n>N$

Figure 2.2.1 : Process Of Bagging (Bootstrap Aggregation)

Boosting (AdaBoost and Gradient Boosting): AdaBoost and Gradient Boosting focus on the correction of errors generated by previous weak models in the ensemble, thus reducing bias and improving predictive accuracy.

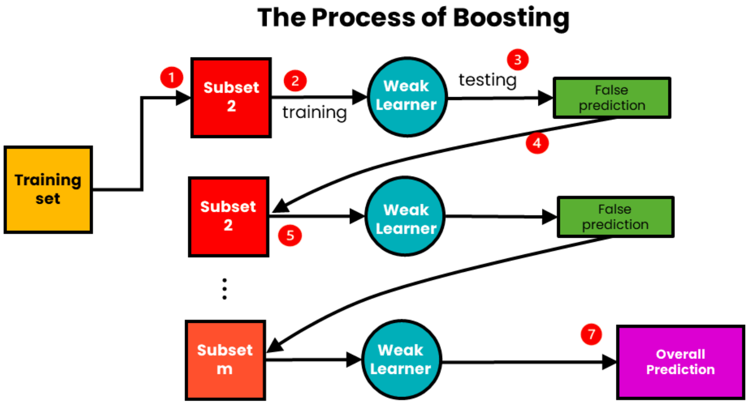


Figure 2.2.2: The process of Boosting

Stacking: A Stacking Classifier uses three main models—Random Forest, AdaBoost, and Gradient Boosting and adds Logistic Regression as the final model. This method tries to make predictions more accurate by using the best parts of different classifiers.

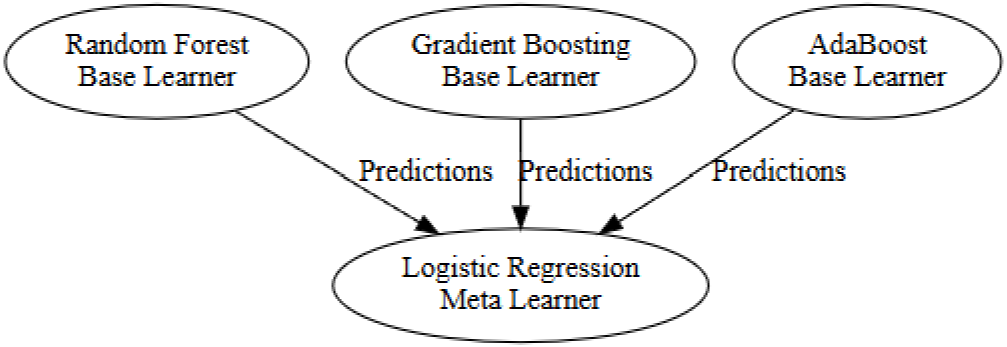


Figure 2.2.3: The process of Stacking Classifier

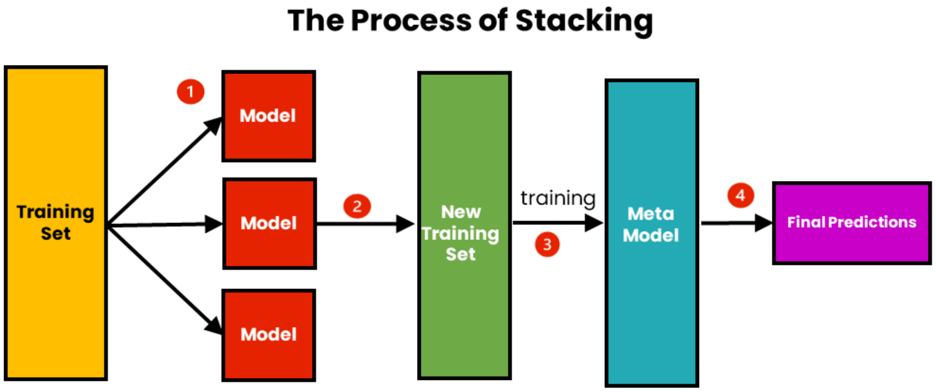


Figure 2.2.4: The process of Stacking Classifier

Voting Classifier: The Voting Classifier takes predictions from Random Forest, AdaBoost, and Gradient Boosting and uses soft voting. Soft voting averages the probabilities from each model.

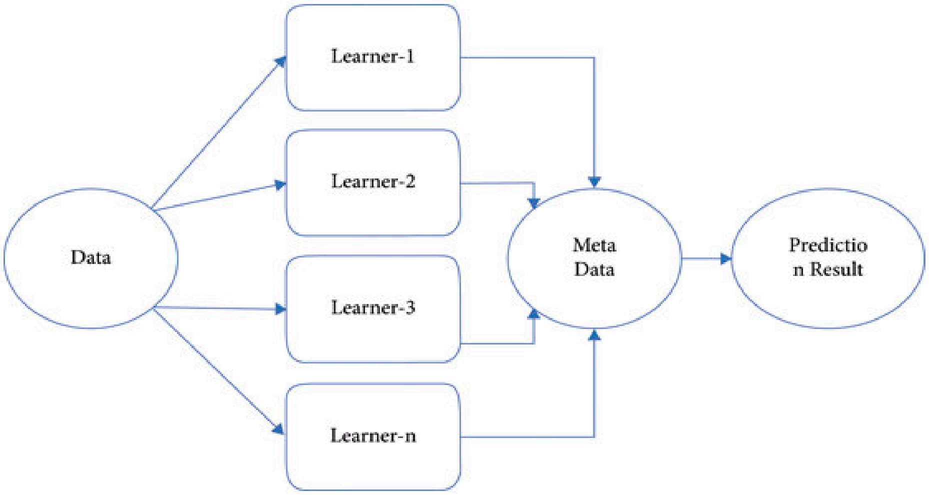


Figure 2.2.5: The process of Voting Classifier

2.3 Explainable AI (XAI) Techniques

To interpret the predictions of the best-performing models, we used two XAI methods: **SHAP** and **LIME**.

SHAP (SHapley Additive exPlanations): SHAP explains how each feature affects the model's output, assigning a "Shapley value" to each feature; hence, it is very clear which features, like packet size and protocol, have the most say in determining whether an activity is "Normal" or "Malicious." SHAP helps us understand the big picture by showing which features matter the most.

We used SHAP in both Random Forest and Gradient Boosting models to understand which features—like packet size, protocol type, or connection duration—were most important in deciding whether a network activity was normal or malicious. Using SHAP, we could get into detail about why some of the data points were tagged as malicious, hence bringing us useful understanding and trust in the model.

LIME (Local Interpretable Model-agnostic Explanations): LIME has been used to explain individual predictions by approximating the complex model with a simpler, interpretable model for specific data points. It helps understand how small changes in features affect a model's decision. LIME thus comes in handy in explaining why a certain network activity was classified as malicious or normal, and it provides local explanations for each prediction.

We used LIME in a separate subroutine to explain specific predictions provided by the Random Forest and Gradient Boosting algorithms. That is, when network activity is malicious, LIME highlighted such features with the greatest influence on that decision. As is illustrated next, by manipulating input features and observing what changes in the model, LIME helped understand what small changes in network properties will result in different classifications of end-use activities. This level of explanation gives an insight into the rationale of the model's decisions regarding specific cases.

Both SHAP and LIME increase the transparency and dependability of models with different levels of interpretability.

2.4 Performance Evaluation

The performance of both models was evaluated based on standard classification metrics such as accuracy, precision, recall, and F1 score. These evaluation metrics give very important insight into the potential of the model in identifying malicious activity while adequately minimizing false positives and false negatives.

- **Accuracy:** The measure of the percent of overall correct identifications, both true positives and true negatives, out of all made predictions.
- **Precision:** The proportion of true positives over the total number of positives predicted, meaning how many of the detected threats are actually malicious.
- **Recall:** True positive ratio concerning the total amount of actual positives, shows the model's efficiency in finding all the real threats.
- **F1 Score:** Precision and recall are synthesized into a harmonic mean, providing an equilibrium assessment of both metrics, which proves particularly beneficial when dealing with imbalanced datasets.

Cross-validation of a model's effectiveness is done by creating folds or subsets from the dataset. Some of the subsets are used for training, and others for measuring performance. Repeating this many times and summing the results will give an overall assessment of the model's performance. The process of cross-validation ensures that the model generalizes well to new, unseen data and therefore avoids overfitting; it gives a better estimation of how the model will perform in real-life applications. This is very important for cybersecurity efforts, as real-world datasets can be quite different from the ones used in training.

3. Results

3.1 Cross-Validation and Performance Evaluation

The following table summarizes the evaluation metrics for each ensemble model:

Table 1: Performance Evaluation of Ensemble Models

Model	Accuracy	Precision	Recall	F1-score
Random Forest	0.526807	0.525622	0.526807	0.525731
AdaBoost	0.510490	0.511104	0.510490	0.510697
Gradient Boosting	0.517483	0.516992	0.517483	0.517167
Stacking	0.491841	0.490985	0.491841	0.491243
Voting Classifier	0.482517	0.483348	0.482517	0.482765

Confusion Matrices:

Confusion matrices for each model are included below to provide further insights into model performance:

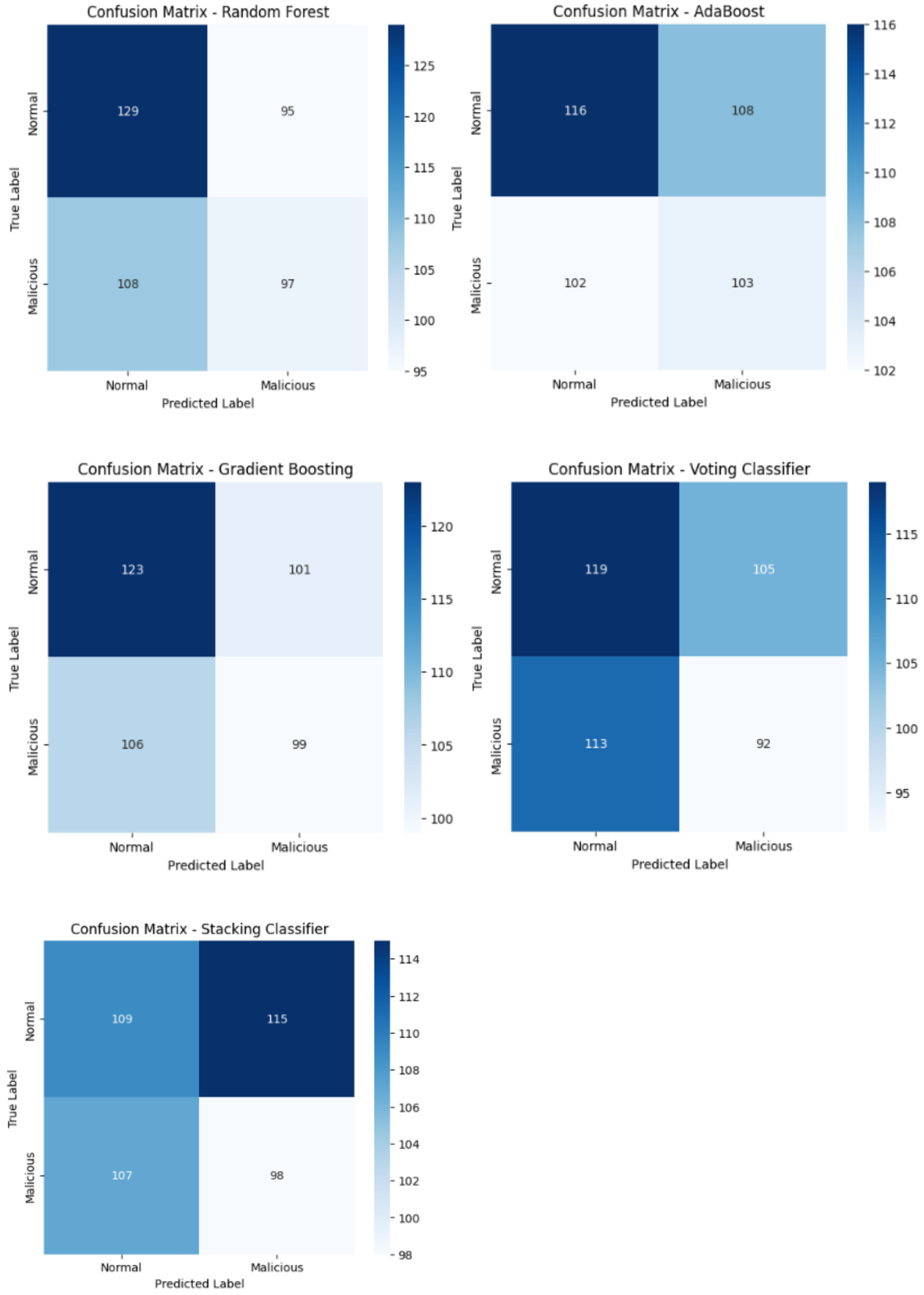


Figure: 3.1.1: Confusion matrix for all the models

3.2 SHAP and LIME Explainability

3.2.1 SHAP Results

SHAP was used to explain the predictions for both Random Forest and Gradient Boosting models. The following SHAP summary plots provide the feature importance for the two modeling methodologies. These plots give a better understanding visually of which particular features have been more influential in predicting the target variable.

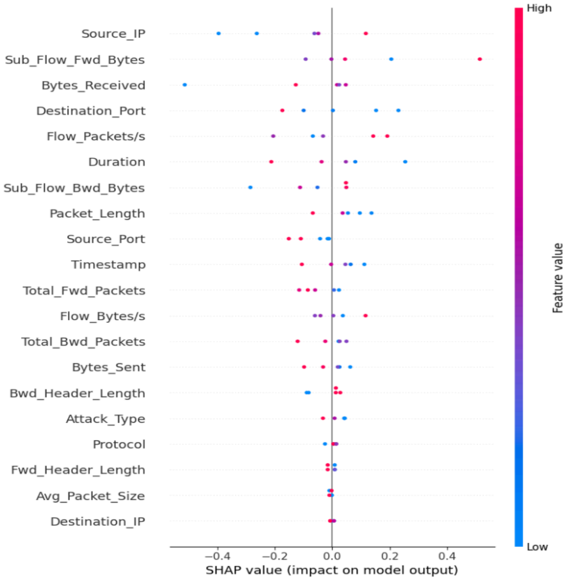


Figure 1: SHAP Summary Plot for Gradient Boosting Model

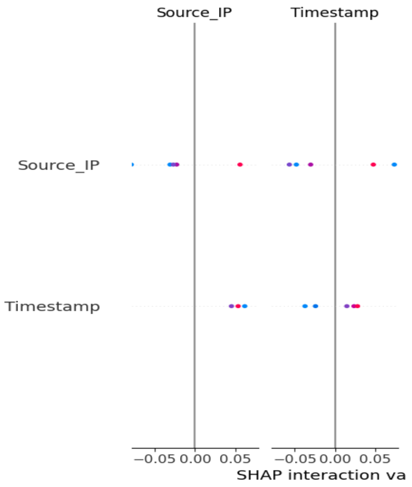


Figure 2: SHAP Summary Plot for Random Forest Model

3.2.2 LIME Results

LIME has been employed for explaining individual predictions. The figure below shows the LIME explanation for a test instance sampled from the data set. LIME highlights the most informative features contributing to the prediction, enabling us to know how the model arrives at a decision concerning an instance.

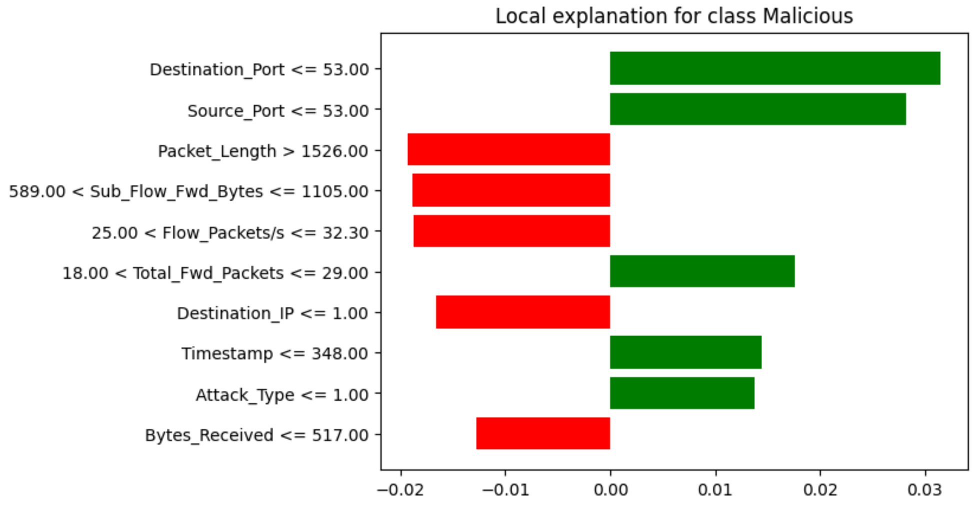


Figure 3: LIME Explanation for a Sample Prediction

4. Discussion

It is from these performances that the algorithmic performances showed that Stacking and Gradient Boosting were the best performers as far as accuracy, precision, recall, and F1-score were concerned, while the other competitive models included Random Forest and Voting Classifier. As for interpretability, SHAP and LIME are quite capable of giving substantial insights into the model's decision-making mechanism. For example, with SHAP it was possible to determine that feature packet size and duration were some of the most important features driving the models' predictions in both the Random Forest and Gradient Boosting models. LIME provided a local explanation of every single prediction, hence highlighting the reason why the respective network activities were classified as malicious or normal. Although ensemble methods improved the overall performance, it was a critical question about model accuracy versus model interpretability. Highly accurate models included Gradient Boosting and Stacking, while they became complex and not so easy to interpret without some XAI techniques.

5. Conclusion

The results of this work serve as proof for two ensemble learning methods, namely Stacking and Gradient Boosting, for detecting cyber threats. The former has outperformed single models quite visibly across various dimensions involving accuracy and other performance metrics. Further improvements were made using SHAP and LIME. Critical insights were derived on feature importance contributing to explaining individual predictions. Further studies might be directed toward searching for additional ensemble methods, optimizing hyperparameter values, and enhancing model interpretability so as to ensure that cybersecurity models remain both accurate and interpretable.