Selina Narain, Neelam Boywah, Zoya Haq
DTSC 870 - Masters Project - Fall 2023
Advisor: Professor Dr. Wenjia Li

Progress Report 7

Timeline: December 6th, 2023 - December 12th, 2023

Accomplishments: What did you accomplish?

Research Topic Idea:

- Comparing machine learning and deep learning algorithms for accuracy and efficiency in detecting malware in android applications.
- Applying FGSM adversarial attack on our highest performance models (Random Forest) for 2 datasets.
- Applying an Adversarial Training Defense Mechanism to bring back up the accuracy and efficiency of the Random Forest Model.

Research:

Adversarial Attacks and Defenses in Deep Learning
https://www.sciencedirect.com/science/article/pii/S209580991930503X

- In the paper there are various algorithms and methods that can be used when creating and implementing the Adversarial Attack.
- Fast Gradient Sign Method used in an Adversarial Attack is an untargeted attack that generates adversarial samples.This one-step attack algorithm executes a one-step update which increases the loss in the steepest direction.
- When FGSM is applied to a targeted attack algorithm (targeted FGSM), it decreases the gradient in the y target label. There can also be a decrease in the cross-entropy if cross-entropy is applied as the loss in the adversarial sample.
- For the defense mechanism, there are different types of adversarial training used.
- The FGSM adversarial training method, trains the model with both benign and FGSM-generated adversarial samples.

DADA Dataset

Based on the research paper, "Debiasing Android Malware Datasets: How can I trust your results if your dataset is biased?"

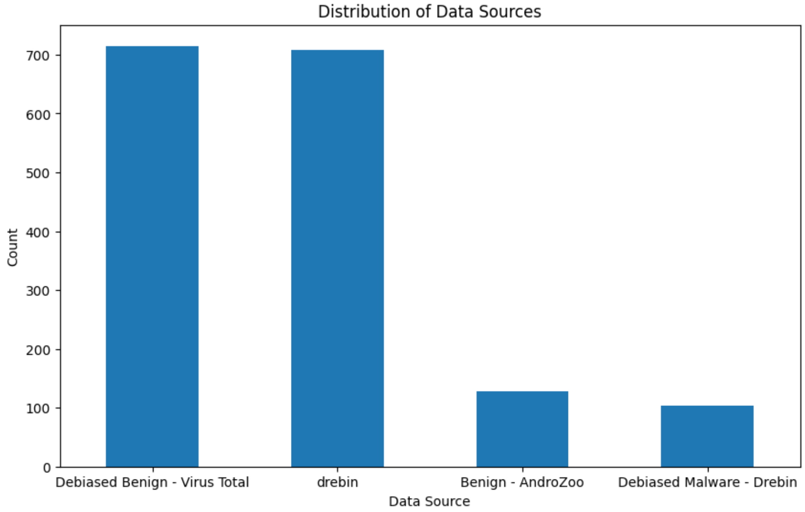
- Contains data sources like AndroZoo, Drebin and AMD.
- The dataset includes debiased data where the samples are malware and benign.
- We see an almost 50/50 data split between benign vs. malware.

- For our research implementation, we utilized the produced mixed dataset which includes debiased processed data from Drebin and AndroZoo.

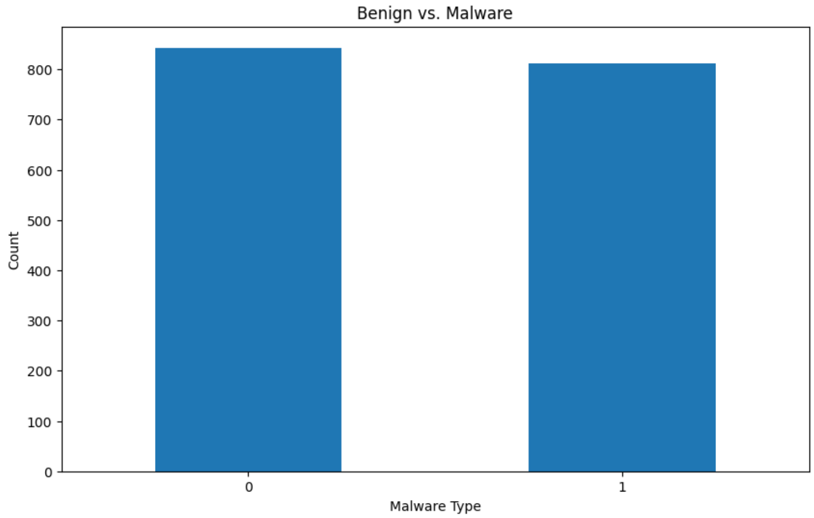
Dada Dataset Implementation

- Imported necessary libraries: numpy, pandas, Sklearn, tensorflow, matplotlib, seaborn.
- Created Visualizations to show the distribution of data sources and benign vs. malware.
- Used feature selection. Defined features and targets, dropped features that weren't going to be used in the training and testing.
- Then we scaled the data using the Standard Scaler function from SKLearn.
- Defined variable classes as 0 and 1. 0 is Benign and 1 is Malware.
- Built and tested 6 machine learning models and 1 deep learning model.
- Obtained their evaluation metrics, classification report, confusion matrix and heatmaps.
- Applied Adversarial Attack on Random Forest Model.
- Applied Defense Mechanism.
- Created comparison of models visualization based on all models built using the DADA dataset.

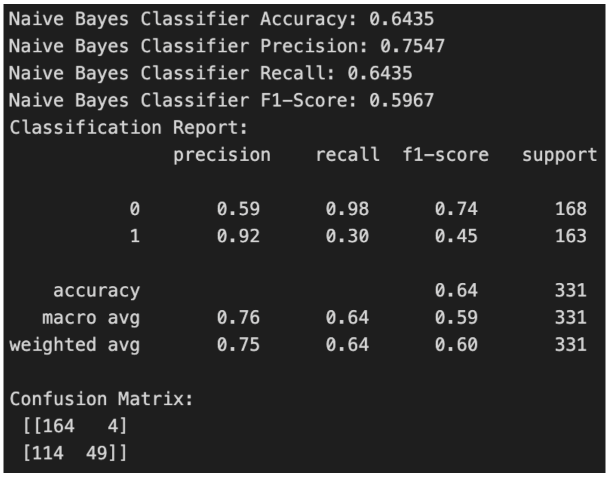
Bar Graph Displaying the Distribution of Data Sources



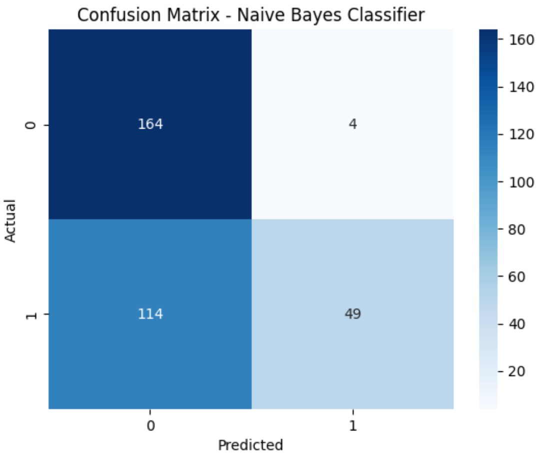
Bar Graph Displaying the distribution of Benign vs. Malware



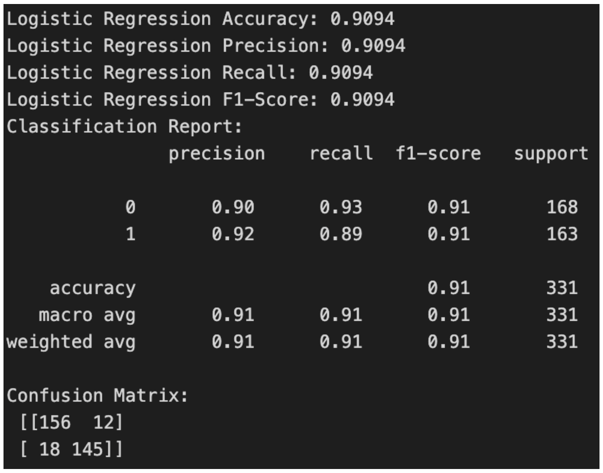
Naive Bayes



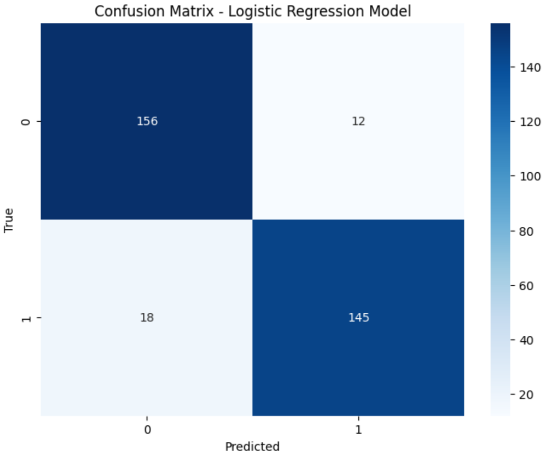
```
Naive Bayes Classifier Accuracy: 0.6435
Naive Bayes Classifier Precision: 0.7547
Naive Bayes Classifier Recall: 0.6435
Naive Bayes Classifier F1-Score: 0.5967
Classification Report:
              precision    recall  f1-score   support

           0       0.59      0.98      0.74       168
           1       0.92      0.30      0.45       163

    accuracy                           0.64       331
   macro avg       0.76      0.64      0.59       331
weighted avg       0.75      0.64      0.60       331

Confusion Matrix:
 [[164   4]
 [114  49]]
```



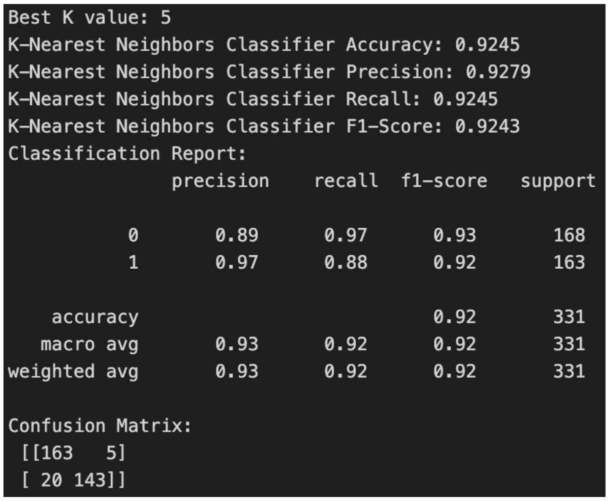
Logistic Regression



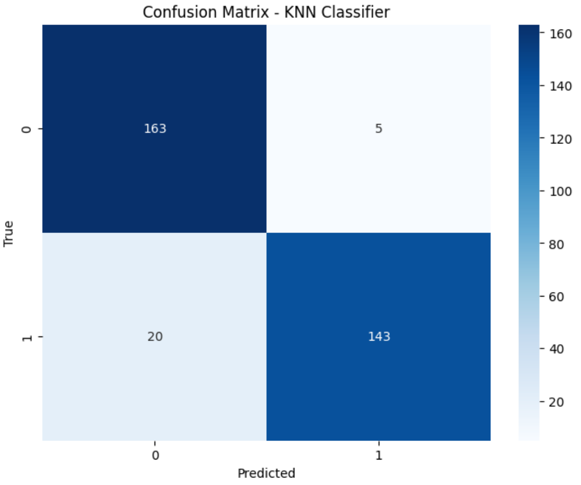
```
Logistic Regression Accuracy: 0.9094
Logistic Regression Precision: 0.9094
Logistic Regression Recall: 0.9094
Logistic Regression F1-Score: 0.9094
Classification Report:
              precision    recall  f1-score   support

           0       0.90      0.93      0.91       168
           1       0.92      0.89      0.91       163

    accuracy                           0.91       331
   macro avg       0.91      0.91      0.91       331
weighted avg       0.91      0.91      0.91       331

Confusion Matrix:
 [[156  12]
 [ 18 145]]
```



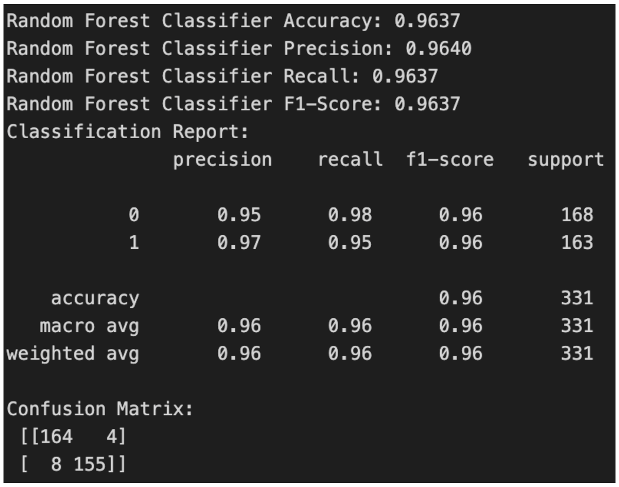
KNN



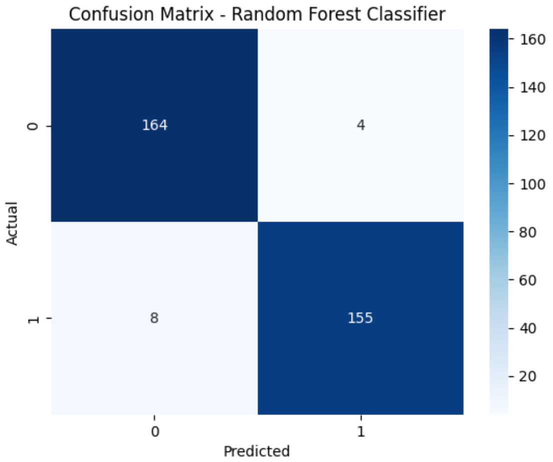
```
Best K value: 5
K-Nearest Neighbors Classifier Accuracy: 0.9245
K-Nearest Neighbors Classifier Precision: 0.9279
K-Nearest Neighbors Classifier Recall: 0.9245
K-Nearest Neighbors Classifier F1-Score: 0.9243
Classification Report:
               precision    recall  f1-score   support

           0       0.89      0.97      0.93       168
           1       0.97      0.88      0.92       163

    accuracy                           0.92       331
   macro avg       0.93      0.92      0.92       331
weighted avg       0.93      0.92      0.92       331

Confusion Matrix:
 [[163   5]
 [ 20 143]]
```



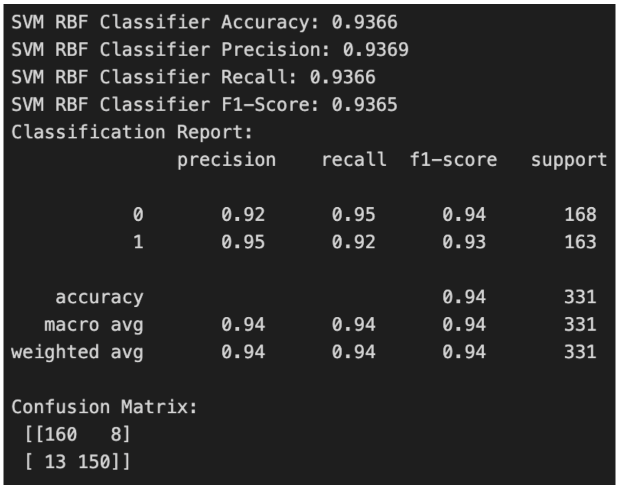
Random Forest



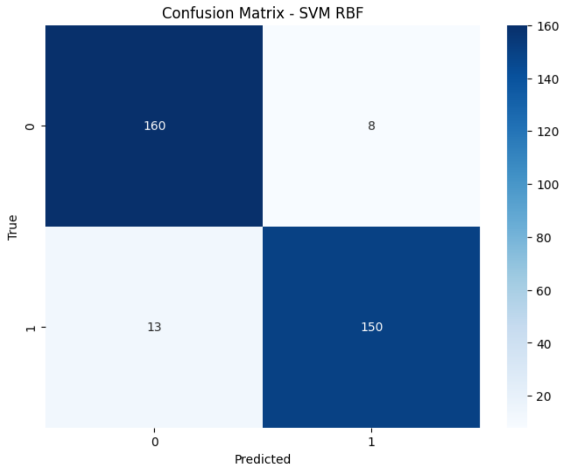
```
Random Forest Classifier Accuracy: 0.9637
Random Forest Classifier Precision: 0.9640
Random Forest Classifier Recall: 0.9637
Random Forest Classifier F1-Score: 0.9637
Classification Report:
               precision    recall  f1-score   support

           0       0.95      0.98      0.96       168
           1       0.97      0.95      0.96       163

    accuracy                           0.96       331
   macro avg       0.96      0.96      0.96       331
weighted avg       0.96      0.96      0.96       331

Confusion Matrix:
 [[164   4]
 [  8 155]]
```



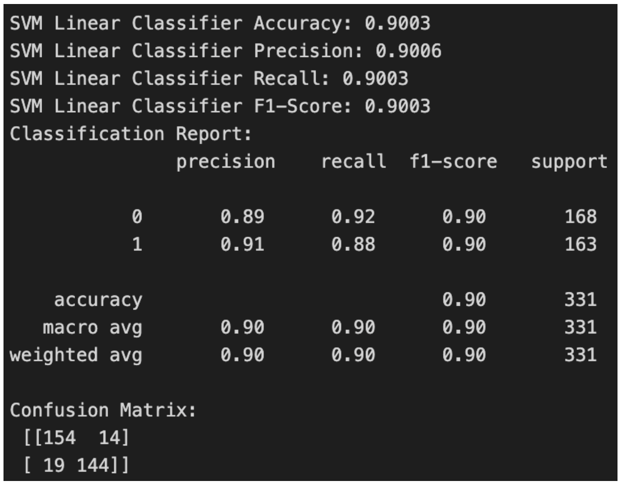
SVM ‘rbf’



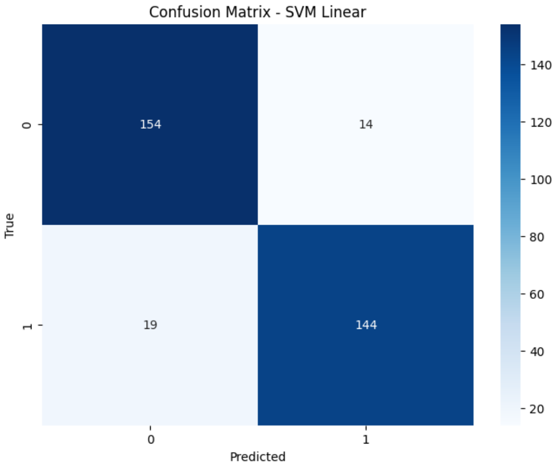
```
SVM RBF Classifier Accuracy: 0.9366
SVM RBF Classifier Precision: 0.9369
SVM RBF Classifier Recall: 0.9366
SVM RBF Classifier F1-Score: 0.9365
Classification Report:
               precision    recall  f1-score   support

           0       0.92      0.95      0.94       168
           1       0.95      0.92      0.93       163

    accuracy                           0.94       331
   macro avg       0.94      0.94      0.94       331
weighted avg       0.94      0.94      0.94       331

Confusion Matrix:
 [[160   8]
 [ 13 150]]
```



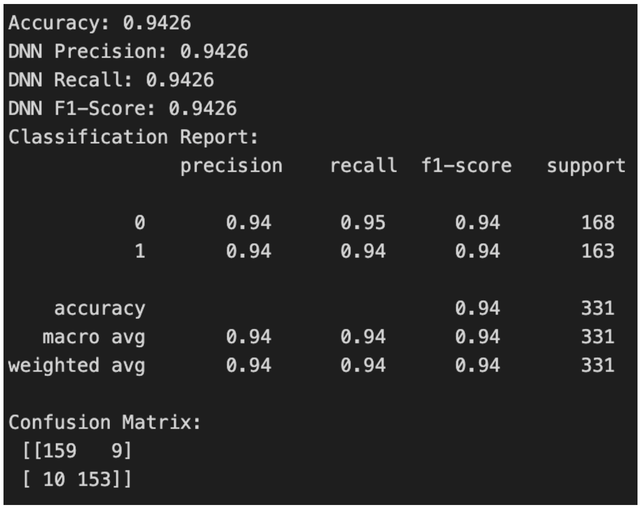
SVM ‘linear’



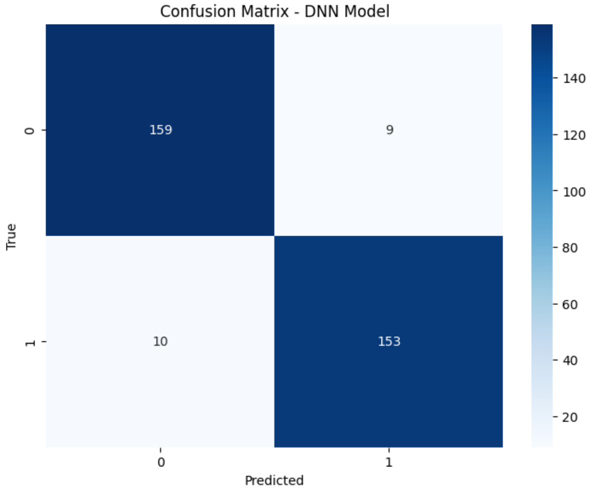
```
SVM Linear Classifier Accuracy: 0.9003
SVM Linear Classifier Precision: 0.9006
SVM Linear Classifier Recall: 0.9003
SVM Linear Classifier F1-Score: 0.9003
Classification Report:
              precision    recall  f1-score   support

           0       0.89      0.92      0.90       168
           1       0.91      0.88      0.90       163

    accuracy                           0.90       331
   macro avg       0.90      0.90      0.90       331
weighted avg       0.90      0.90      0.90       331

Confusion Matrix:
 [[154  14]
 [ 19 144]]
```



Dense Neural Network (DNN)



```
Accuracy: 0.9426
DNN Precision: 0.9426
DNN Recall: 0.9426
DNN F1-Score: 0.9426
Classification Report:
              precision    recall  f1-score   support

           0       0.94      0.95      0.94       168
           1       0.94      0.94      0.94       163

    accuracy                           0.94       331
   macro avg       0.94      0.94      0.94       331
weighted avg       0.94      0.94      0.94       331

Confusion Matrix:
 [[159   9]
 [ 10 153]]
```



Implementation on New Brunswick: CICMaldroid 2020 Dataset”

- Fine-tuned the adversarial attack and defense mechanism.
- Created classification reports and confusion matrices for all machine learning and deep learning models.

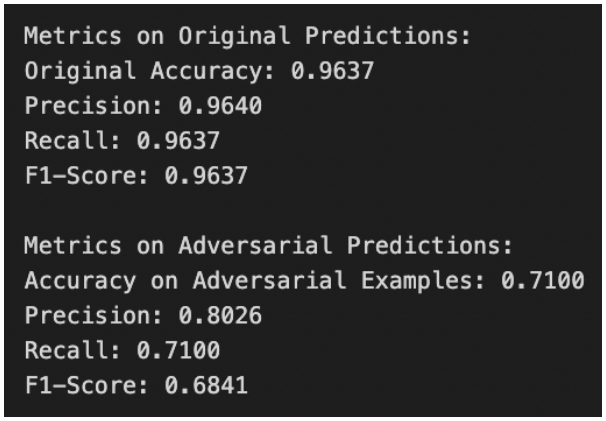
Analysis on DADA Dataset:

- The Highest Performance model is Random Forest.
- Lowest Performance model is Naive Bayes.
- Second highest performing model is Dense Neural Network.
- With the FGSM Adversarial attack and defense mechanism, we were able to attack the Random Forest model successfully as well as defend it. The adversarial attack dropped the model's performance down to a 71% accuracy.

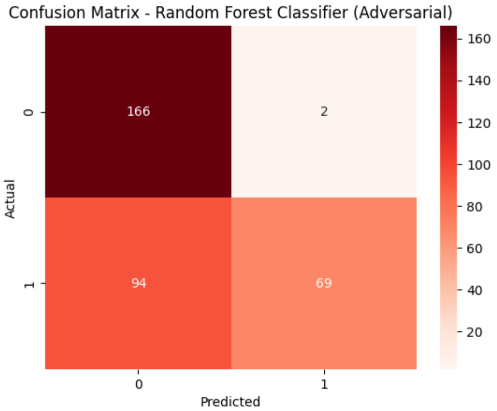
When applying the defense mechanism, we were able to bring back up the model's accuracy to 77.5%.

- Based on our evaluation metrics, we saw that because the DADA dataset contains debiased data and is more robust, it was harder to attack the Random Forest model and defend it.

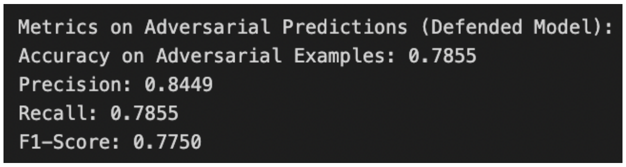
Adversarial Attack (FGSM Attack) (DADA Dataset)



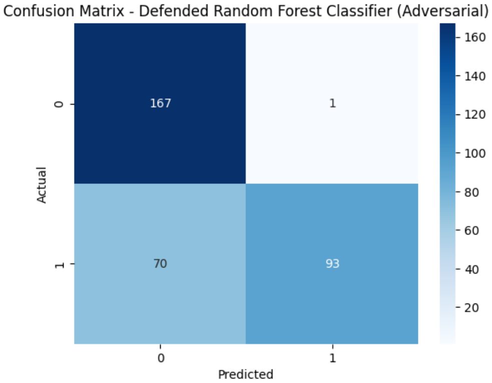
```
Metrics on Original Predictions:
Original Accuracy: 0.9637
Precision: 0.9640
Recall: 0.9637
F1-Score: 0.9637

Metrics on Adversarial Predictions:
Accuracy on Adversarial Examples: 0.7100
Precision: 0.8026
Recall: 0.7100
F1-Score: 0.6841
```



Defense Mechanism (DADA Dataset)



```
Metrics on Adversarial Predictions (Defended Model):
Accuracy on Adversarial Examples: 0.7855
Precision: 0.8449
Recall: 0.7855
F1-Score: 0.7750
```

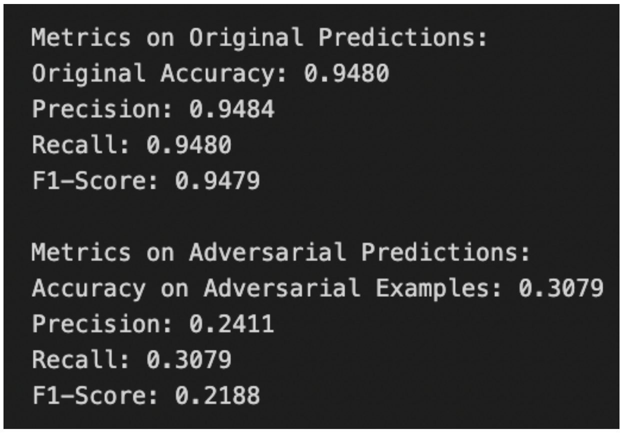


Analysis on New Brunswick: CICMaldroid 2020 Dataset:

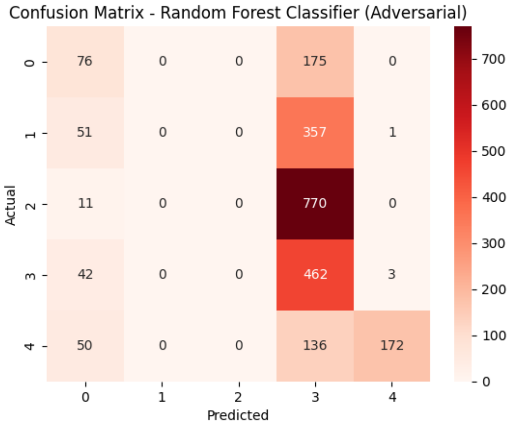
- Similarly to the DADA dataset, we applied the Adversarial attack using Fast Gradient Sign Method (FGSM) on the Random Forest Model and Adversarial Training as a Defense Mechanism. We were able to adjust the adversarial attack and defense mechanism based on this dataset to then see how the Random Forest model would perform.
- The adversarial attack significantly dropped the Random Forest models accuracy down to a 30.79% and the defense mechanism brought the models performance back up to an accuracy of 89.16%.

- In this CICMaldroid 2020 Dataset, we see more of a difference in metrics with this attack and defense compared to the DADA dataset.

Adversarial Attack (FGSM Attack) (New Brunswick: CICMaldroid 2020 Dataset)



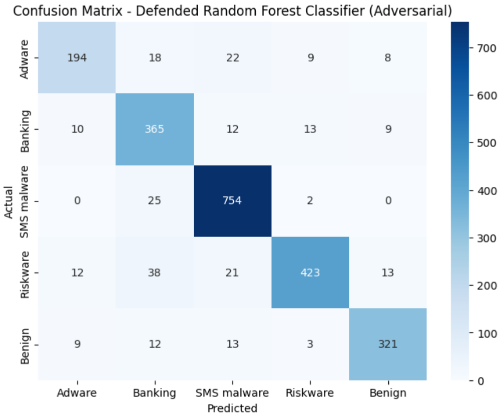
```
Metrics on Original Predictions:
Original Accuracy: 0.9480
Precision: 0.9484
Recall: 0.9480
F1-Score: 0.9479

Metrics on Adversarial Predictions:
Accuracy on Adversarial Examples: 0.3079
Precision: 0.2411
Recall: 0.3079
F1-Score: 0.2188
```



Confusion Matrix - Random Forest Classifier (Adversarial)

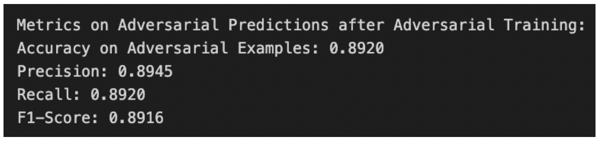
Actual \ Predicted	0	1	2	3	4
0	76	0	0	175	0
1	51	0	0	357	1
2	11	0	0	770	0
3	42	0	0	462	3
4	50	0	0	136	172

Defense Mechanism (New Brunswick: CICMaldroid 2020 Dataset)



Confusion Matrix - Defended Random Forest Classifier (Adversarial)

Actual \ Predicted	Adware	Banking	SMS malware	Riskware	Benign
Adware	194	18	22	9	8
Banking	10	365	12	13	9
SMS malware	0	25	754	2	0
Riskware	12	38	21	423	13
Benign	9	12	13	3	321

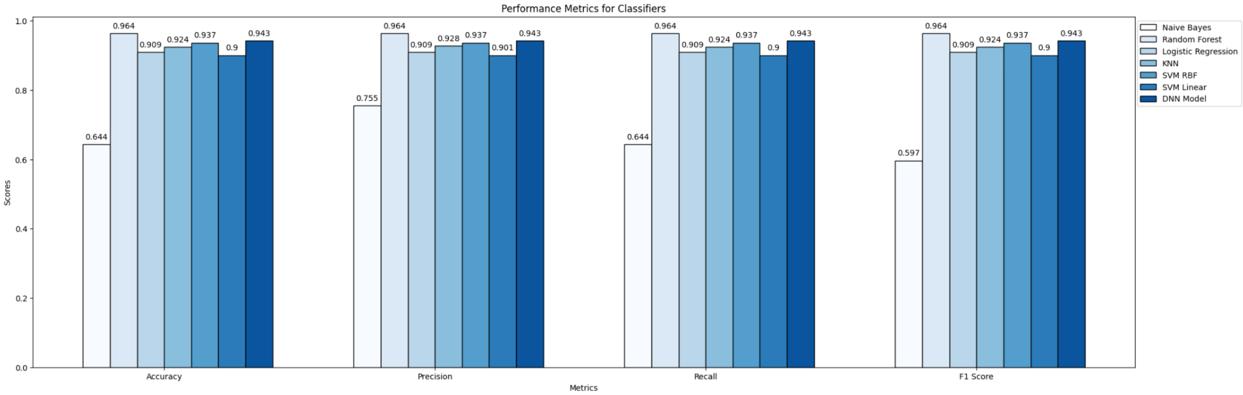


```
Metrics on Adversarial Predictions after Adversarial Training:
Accuracy on Adversarial Examples: 0.8920
Precision: 0.8945
Recall: 0.8920
F1-Score: 0.8916
```

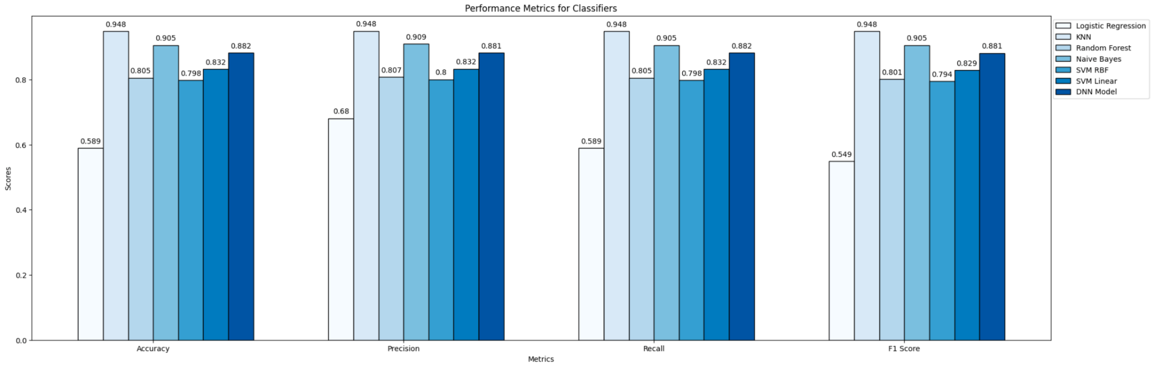
Updated Comparisons of Models Performance Visualization

- They have their accuracy score displayed on top of the bars for better understanding.

Models Performance - DADA Dataset

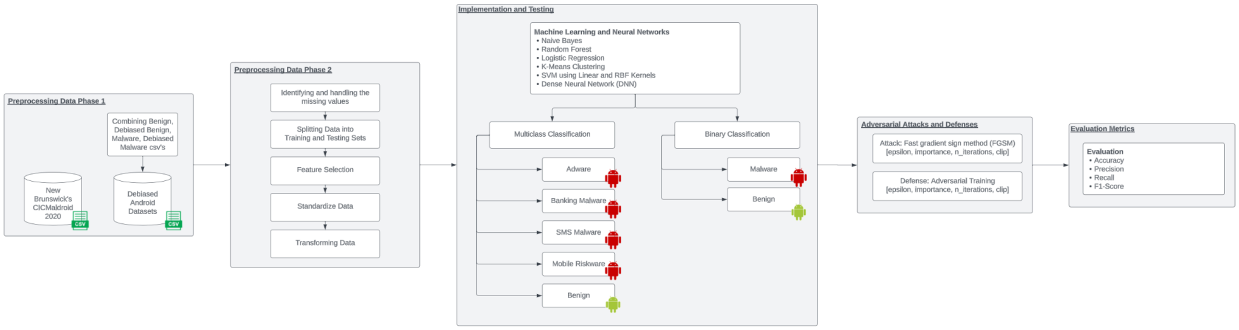


Models Performance - New Brunswick: CICMaldroid 2020 Dataset



Projected FlowChart

- Updated the Flowchart that shows the 5 steps of our research project:
  - Preprocessing Data Phase 1
  - Preprocessing Data Phase 2
  - Implementation Testing
  - Adversarial Attack and Defense Mechanisms
  - Evaluation Metrics.


Preprocessing Data Phase 1
Combining Benign, Debiased Benign, Malware, Debiased Malware csv's
New Brunswick's CICMaldroid 2020
Debiased Android Datasets
CSV
CSV
Preprocessing Data Phase 2
Identifying and handling the missing values
Splitting Data into Training and Testing Sets
Feature Selection
Standardize Data
Transforming Data
Implementation and Testing
Machine Learning and Neural Networks
• Naive Bayes
• Random Forest
• Logistic Regression
• K-Means Clustering
• SVM using Linear and RBF Kernels
• Dense Neural Network (DNN)
Multiclass Classification
Adware
Banking Malware
SMS Malware
Mobile Riskware
Benign
Binary Classification
Malware
Benign
Adversarial Attacks and Defenses
Attack: Fast gradient sign method (FGSM) [epsilon, importance, n_iterations, clip]
Defense: Adversarial Training [epsilon, importance, n_iterations, clip]
Evaluation Metrics
Evaluation
• Accuracy
• Precision
• Recall
• F1-Score

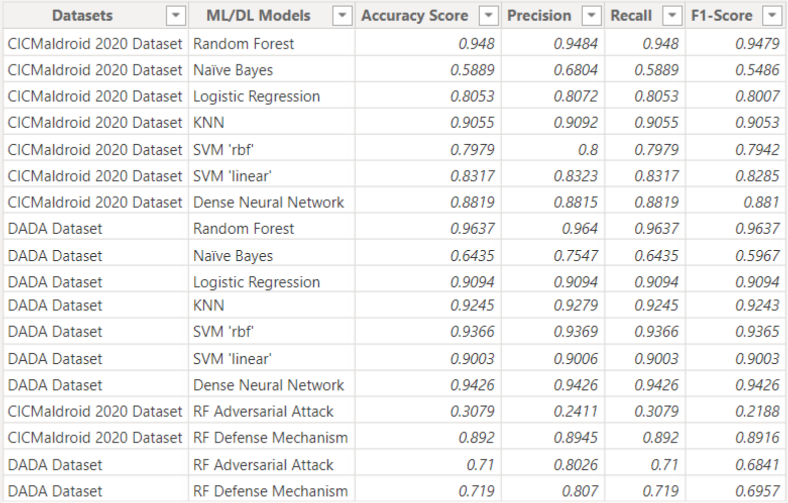
Adversarial Attack and Defense Mechanism Metrics

		Accuracy Score	Precision	Recall	F1-Score
CICMaldroid 2020 Dataset	RF Adversarial Attack	0.9480	0.9484	0.9480	0.9479
	RF Defense Mechanism	0.8920	0.8945	0.8920	0.8916
DADA Dataset	RF Adversarial Attack	0.7100	0.8026	0.7100	0.6841
	RF Defense Mechanism	0.7190	0.8070	0.7190	0.6957

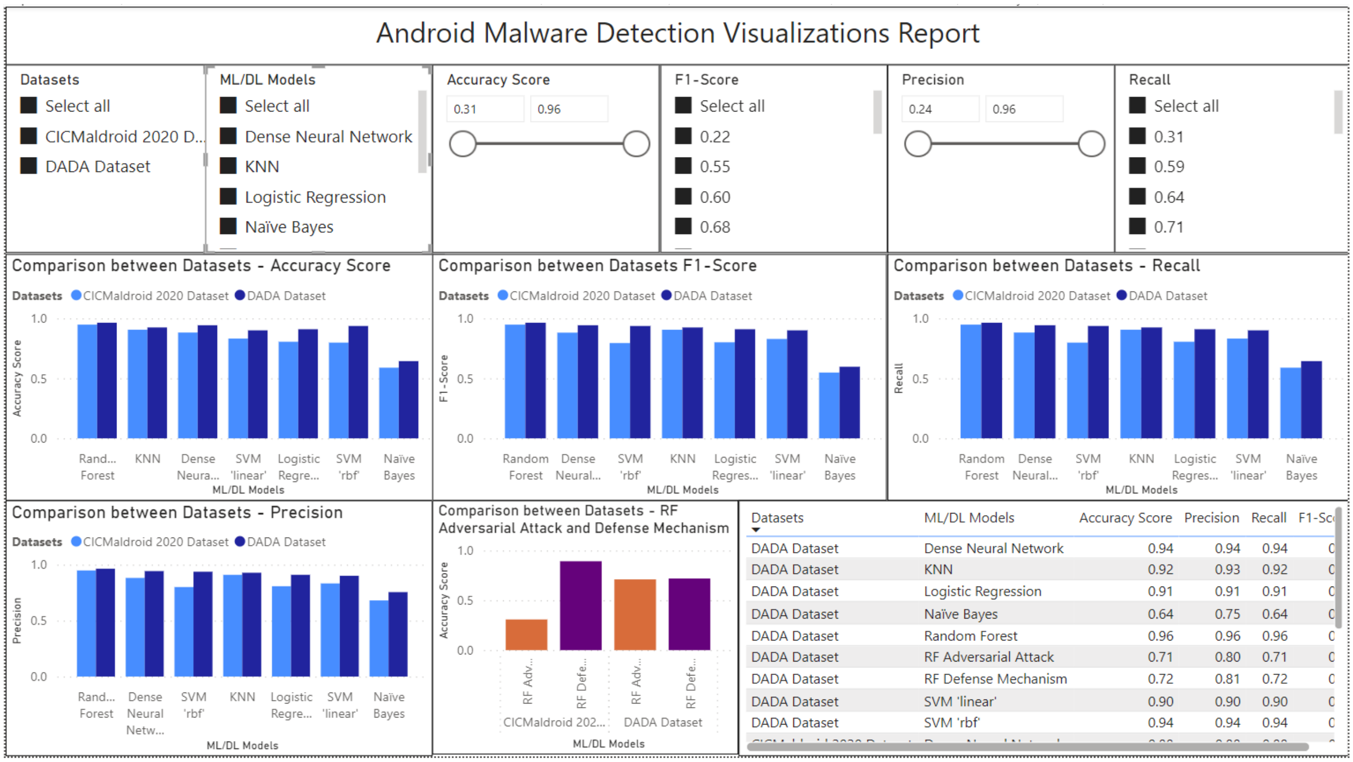
Comparisons between datasets for RF Adversarial Attack and Defense Mechanisms

Updated Power BI Report for Presentation

- Report includes 2 datasets: New Brunswick: CICMaldroid 2020 Dataset and DADA Dataset.
- Report also shows the metrics for the adversarial attack and defense mechanism on the Random Forest model for both datasets.
- Created a new visualization bar graph to compare the adversarial attack and defense mechanism metrics on both datasets.



Datasets	ML/DL Models	Accuracy Score	Precision	Recall	F1-Score
CICMaldroid 2020 Dataset	Random Forest	0.948	0.9484	0.948	0.9479
CICMaldroid 2020 Dataset	Naïve Bayes	0.5889	0.6804	0.5889	0.5486
CICMaldroid 2020 Dataset	Logistic Regression	0.8053	0.8072	0.8053	0.8007
CICMaldroid 2020 Dataset	KNN	0.9055	0.9092	0.9055	0.9053
CICMaldroid 2020 Dataset	SVM 'rbf'	0.7979	0.8	0.7979	0.7942
CICMaldroid 2020 Dataset	SVM 'linear'	0.8317	0.8323	0.8317	0.8285
CICMaldroid 2020 Dataset	Dense Neural Network	0.8819	0.8815	0.8819	0.881
DADA Dataset	Random Forest	0.9637	0.964	0.9637	0.9637
DADA Dataset	Naïve Bayes	0.6435	0.7547	0.6435	0.5967
DADA Dataset	Logistic Regression	0.9094	0.9094	0.9094	0.9094
DADA Dataset	KNN	0.9245	0.9279	0.9245	0.9243
DADA Dataset	SVM 'rbf'	0.9366	0.9369	0.9366	0.9365
DADA Dataset	SVM 'linear'	0.9003	0.9006	0.9003	0.9003
DADA Dataset	Dense Neural Network	0.9426	0.9426	0.9426	0.9426
CICMaldroid 2020 Dataset	RF Adversarial Attack	0.3079	0.2411	0.3079	0.2188
CICMaldroid 2020 Dataset	RF Defense Mechanism	0.892	0.8945	0.892	0.8916
DADA Dataset	RF Adversarial Attack	0.71	0.8026	0.71	0.6841
DADA Dataset	RF Defense Mechanism	0.719	0.807	0.719	0.6957



Android Malware Detection Visualizations Report

Datasets
■ Select all
■ CICMaldroid 2020 D...
■ DADA Dataset

ML/DL Models
■ Select all
■ Dense Neural Network
■ KNN
■ Logistic Regression
■ Naïve Bayes

Accuracy Score
0.31 0.96

F1-Score
■ Select all
■ 0.22
■ 0.55
■ 0.60
■ 0.68

Precision
0.24 0.96

Recall
■ Select all
■ 0.31
■ 0.59
■ 0.64
■ 0.71

Datasets	ML/DL Models	Accuracy Score	Precision	Recall	F1-Sc
DADA Dataset	Dense Neural Network	0.94	0.94	0.94	0
DADA Dataset	KNN	0.92	0.93	0.92	0
DADA Dataset	Logistic Regression	0.91	0.91	0.91	0
DADA Dataset	Naïve Bayes	0.64	0.75	0.64	0
DADA Dataset	Random Forest	0.96	0.96	0.96	0
DADA Dataset	RF Adversarial Attack	0.71	0.80	0.71	0
DADA Dataset	RF Defense Mechanism	0.72	0.81	0.72	0
DADA Dataset	SVM 'linear'	0.90	0.90	0.90	0
DADA Dataset	SVM 'rbf'	0.94	0.94	0.94	0

Upcoming Plan: What do you plan to do in upcoming weeks?

- Completing research paper
- Completing final report
- Completing the presentation

Obstacles & Concerns: Were there any obstacles or barriers that prevented you from getting things done?

- What format is the research paper and final report supposed to be? APA? IEEE
- Can the final report (5 copies) be printed back and front?]
- Flowchart: The Presentation slides have a horizontal layout and the final report has the vertical layout. Is that okay?
- Literature Review in presentation slides