Assignment: 3

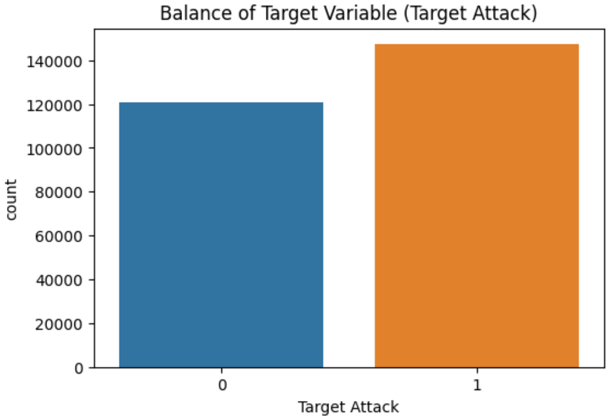
Mohamed Hany Mohamed Sabry (300389913)

Part1: Static model

Data Analysis:

Dataset Overview: The dataset, referred to as 'Static_dataset.csv', consists of 268,074 entries with 16 columns, including both numerical and categorical data types.

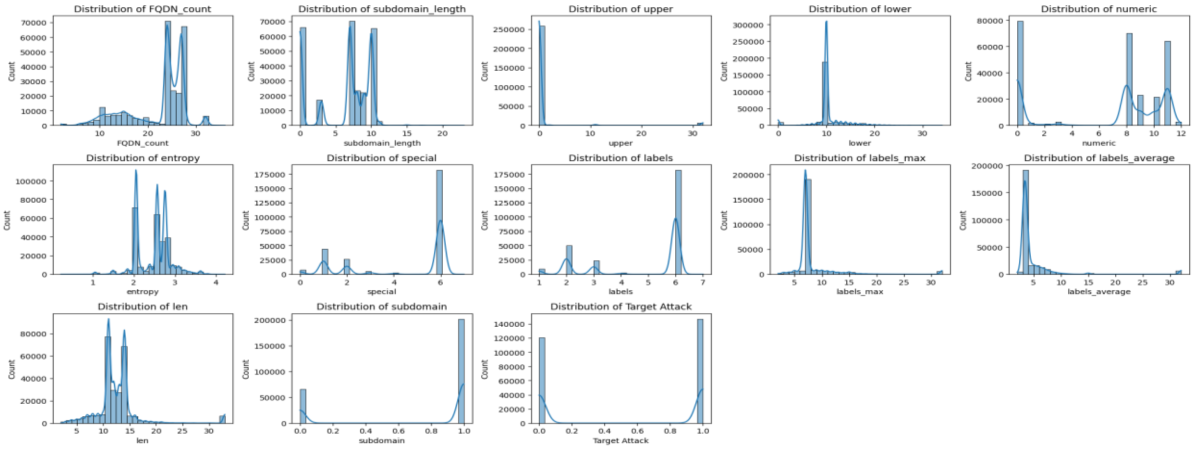
Data Imbalance Analysis: The 'Target Attack' variable showed a relatively balanced distribution, with approximately 55% labeled as '1' (attack) and 45% labeled as '0' (no attack), indicating no significant imbalance.



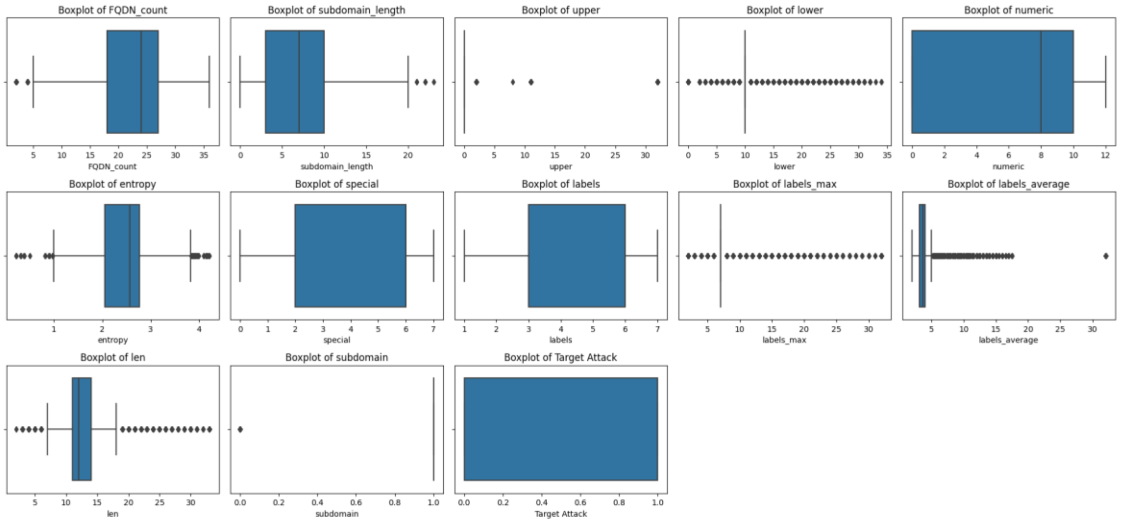
Statistical Analysis:

1. **Correlation Matrix**: A heatmap was generated to examine the correlations between numerical features. It revealed significant correlations that could influence model predictions, like the strong relationship between 'numeric' and 'FQDN_count'.

2. **Distribution Analysis**: Histograms were created for each numerical feature to assess their distribution characteristics. This analysis helped identify features with skewed distributions, which might require transformation to improve model performance.



3. **Outlier Detection**: Boxplots for each numerical feature were constructed to detect outliers. This step was crucial for identifying extreme values that could potentially distort the model's performance.

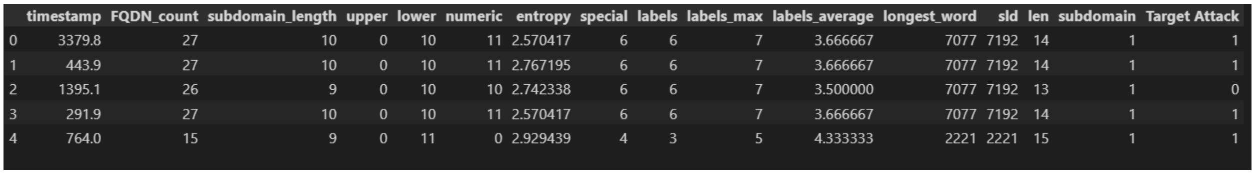


4. **Skewness Detection**: The skewness of each numerical feature was calculated to quantify the asymmetry of their distribution. Features with high skewness were noted, as they might benefit from normalization or other transformations.

Data Cleansing and Feature Engineering

During data preprocessing, the 'longest_word' attribute's missing values were filled with the mode, timestamps were transformed into numerical seconds, and categorical features like 'longest_word' and 'sld' were hashed and normalized to ensure uniformity and model compatibility. This meticulous preparation of the dataset was crucial for the subsequent feature selection and modeling phases, setting the stage for the successful application of machine learning algorithms.

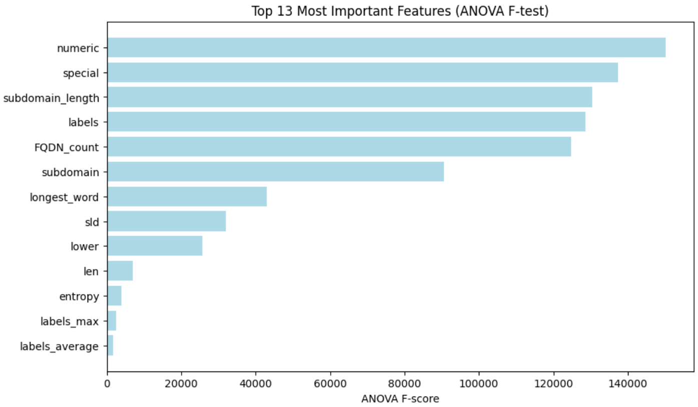
Figure showing results after Data Cleansing and Feature Engineering:

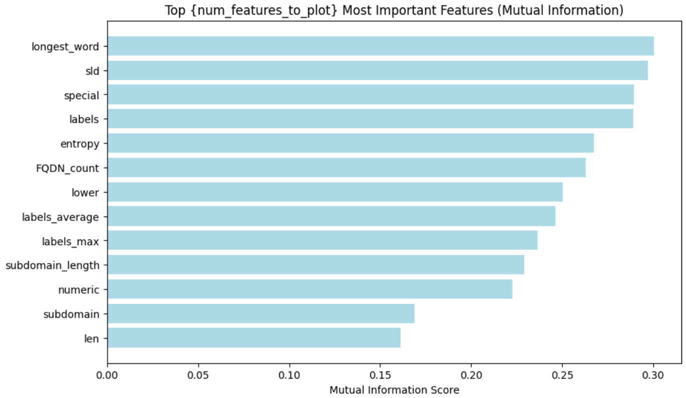


	timestamp	FQDN_count	subdomain_length	upper	lower	numeric	entropy	special	labels	labels_max	labels_average	longest_word	sld	len	subdomain	Target Attack
0	3379.8	27	10	0	10	11	2.570417	6	6	7	3.666667	7077	7192	14	1	1
1	443.9	27	10	0	10	11	2.767195	6	6	7	3.666667	7077	7192	14	1	1
2	1395.1	26	9	0	10	10	2.742338	6	6	7	3.500000	7077	7192	13	1	0
3	291.9	27	10	0	10	11	2.570417	6	6	7	3.666667	7077	7192	14	1	1
4	764.0	15	9	0	11	0	2.929439	4	3	5	4.333333	2221	2221	15	1	1

Feature Selection

Three feature selection methods were used: Mutual Information, ANOVA F-test, and Chi-squared, each highlighting key features like 'longest_word' and 'numeric'. These methods offered a thorough approach to identifying predictive features, which contributed to the high F1-Scores of the trained models.

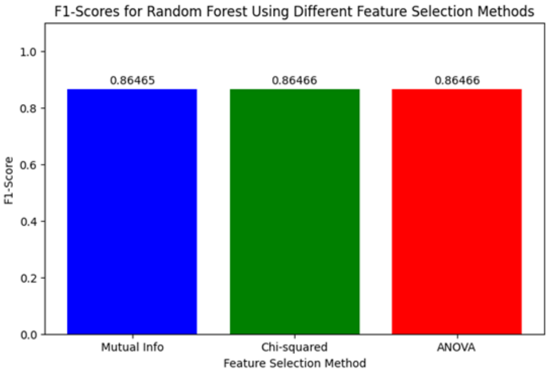


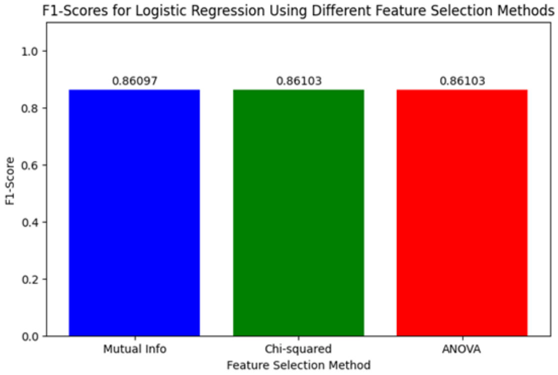


Model Training and Evaluation

The model training and evaluation involved Random Forest and Logistic Regression, chosen for their suitability for binary classification. Random Forest was favored for its handling of complex data, while Logistic Regression was used for its simplicity and interpretability. Both models were assessed using the F1-Score to gauge their precision-recall balance. Ultimately, Random Forest, paired with Chi-squared feature selection, yielded the best performance, as indicated by the highest F1-Score.

These figures show evaluations for two selected models:





Data

Splitting: A 70-30 train-test split was utilized, offering a balance between sufficient training data and a substantial validation set, which is a standard practice in machine learning to ensure robust model evaluation while avoiding overfitting.

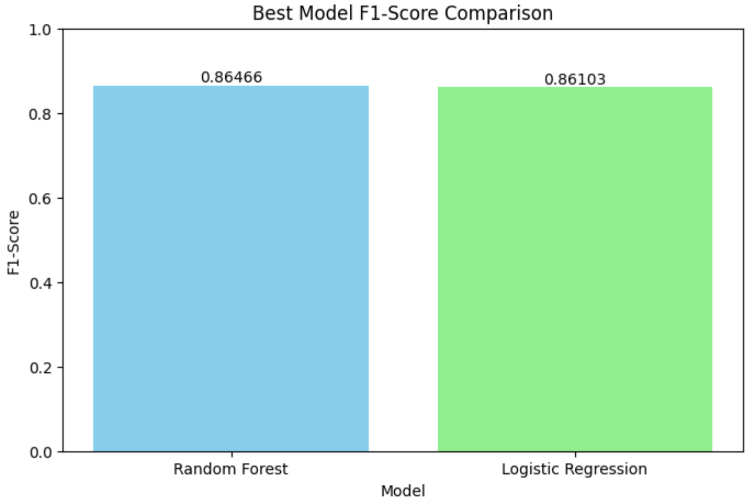
Performance Metrics: The F1-Score was chosen for its balanced approach to measuring precision and recall, crucial for the high-stakes task of predicting attacks where both false alarms and missed detections are costly. It is robust against class imbalance and aligns with the security-focused objectives of the task.

Hyperparameter Tuning and Model Optimization

Optimal hyperparameters for the Random Forest model included 100 trees (n_estimators), a maximum tree depth of 20 (max_depth), and the use of entropy for the quality of splits (criterion). This tuning, along with a MinMaxScaler in the pipeline, led to a refined model with an F1-Score of 0.865, indicating effective tuning for the task.

Results and Discussion

Model Comparison: the Random Forest model, coupled with Chi-squared feature selection, achieved an F1-Score of 0.86466. This was slightly higher than the Logistic Regression, which scored an F1-Score of 0.86103 using the same feature selection technique, highlighting Random Forest's slightly better performance in this specific predictive task.



Feature Selection Impact: Chi-squared feature selection method consistently provided better results for both models.

Part2: Dynamic model

Data Handling: Windows of 1,000 Data Points

Data is efficiently processed in batches of 1,000 records from a Kafka stream, balancing real-time data handling with manageability.

Training Reevaluation Process

The model is reevaluated based on a performance threshold (F1 score < 0.85), triggering retraining with both historical and new data. This continuous learning approach is key in adapting to new and old attack patterns.

Setting the 0.85 F1 score threshold for dynamic model retraining aligns with the proven performance of the static model, ensuring consistency.

Selection and Justification of Performance Metrics

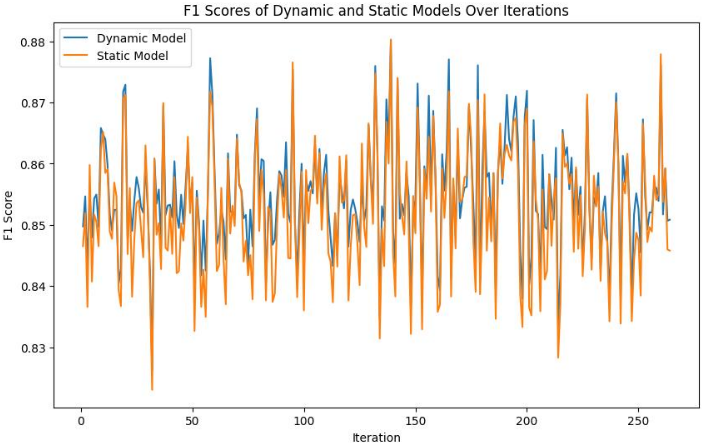
The F1 score, balancing precision and recall is aptly chosen for its relevance in scenarios where both false positives and false negatives are critical.

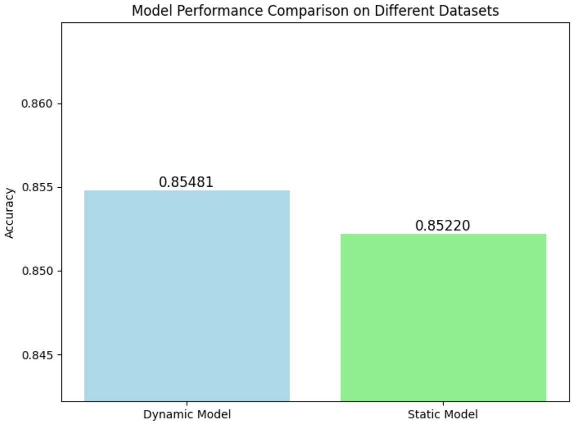
Model Evaluation

Both static and dynamic models are compared using the F1 score, providing a fair assessment of the dynamic model's adaptability over time.

Results and Analysis

Results are effectively visualized, showing the dynamic model's performance across iterations. The model demonstrates adaptability, often outperforming the static model post-retraining.





Advantages

Adaptability: The dynamic model excels in adjusting to new threats, essential in the ever-evolving cybersecurity landscape.

Continuous Learning: It incorporates both new and historical data, ensuring comprehensive learning.

Balanced Evaluation: The use of the F1 score ensures a nuanced assessment of the model's precision and recall.

Limitations

Resource Intensity: Continuous retraining could demand significant computational resources.

Data Dependency: The model's performance is closely tied to the quality and representation of the input data.

Key Learnings

Dynamic Learning in Cybersecurity: The project underscores the necessity for models that can quickly adapt to changing threat patterns.

Importance of Performance Metrics: It highlights the critical role of selecting appropriate metrics like the F1 score for balanced evaluation.

Efficient Resource Management: The project points to the need for effective data and resource management strategies in handling large-scale data