Visual Analysis Of Network Security Datasets

Sai Sreeja Singireddy
Masters in Computer and Information Sciences
University of Houston,Main Campus
Houston, TX
ssingir2@cougarnet.uh.edu

Sai Sri Charishma Vaddi
Masters in Computer and Information Sciences
University of Houston,Main Campus
Houston, TX
svaddi2@cougarnet.uh.edu

***Abstract*—Class imbalance and overlap are critical issues that can significantly impact the performance of data-driven models such as machine learning classifiers.In order to identify these, extensive visual analysis was conducted on the UNSW-NB15 and KDD Cup 1999 datasets, which are common obstacles in the field of network security. The use of various preprocessing techniques and advanced visualization methods, such as PCA, t-SNE, Kmeans, UMAP, and Parallel Coordinates, has led to the development of a Visualization Dashboard. This Dashboard provides a deeper understanding of the data's structure by highlighting areas of class imbalance and overlap, which are essential for refining the development of robust and accurate machine learning models. Through this analysis, the research delivers crucial insights into the dataset characteristics, aiding in the design of more effective preprocessing strategies and the selection of appropriate machine learning algorithms that can handle these challenges.**

***Index Terms*—Intrusion detection dataset, dataset preprocessing, class imbalance, class overlap, Visualization Dashboard**

I. Introduction

Class imbalance and overlap present significant challenges in data mining, particularly affecting classifier model performance in fields like cybersecurity and medicine. Class imbalance occurs when certain categories are underrepresented, potentially causing classifiers to favor the majority class. Class overlap complicates matters further by having different categories with similar features, making it difficult for algorithms to distinguish between them. Despite their common use in research, datasets like UNSW-NB15 and KDD Cup 1999 still struggle with these issues, which can skew analysis outcomes.

Visualization is a pivotal tool in data analysis, particularly when preparing datasets for the development of data-driven classifier models, such as those used in intrusion detection. It transforms complex data into comprehensible visual formats, thereby aiding in the communication of critical information and the identification of patterns that are vital for constructing optimized models.

To address these challenges, a detailed preprocessing and visualization approach is essential. By refining datasets through techniques such as PCA, which simplifies data to lower dimensions, and employing a suite of visualization tools—3D scatter plots, t-SNE, K-means clustering, and UMAP—we can reveal and subsequently address class imbalances and overlaps. Such visual analyses are pivotal, offering scalable, interpretable, and effective methods for illustrating and resolving issues that hinder accurate data classification. Furthermore, the integration of these techniques into an interactive visualization dashboard allows for a dynamic and comprehensive examination of the data, empowering users to tailor the analysis to specific needs and derive deeper insights.

II. Literature Review

In the realm of cybersecurity, the role of visualization techniques has become increasingly pivotal, as evidenced by a range of studies focusing on the intricate challenges of representing complex security data. Cappers et al. (2018), in their presentation at the IEEE Symposium on Visualization for Cyber Security (VizSec), introduced ”Eventpad” – a tool that epitomizes the utilization of visual analytics for rapid malware analysis and reverse engineering. This work not only underscores the urgency of visual tools in deciphering complex cybersecurity data but also illustrates how such tools can significantly expedite the understanding and analysis of malware.

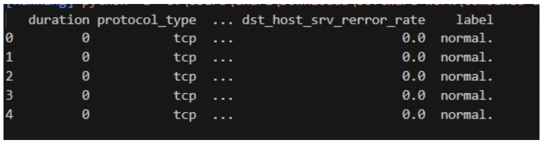
The studies by Ruan et al. focus on visualizing large-scale security data, emphasizing the KDD99 cup dataset and extensive network traffic. Their work highlights the importance of visual tools in detecting patterns and anomalies in network data, key for cybersecurity. Zong, Chow, and Susilo contribute by specializing these visualization techniques for network intrusion detection, underscoring their relevance in handling complex security data.

Collectively, these studies form a foundation that underscores the indispensability of advanced visualization in the analysis of cybersecurity data. They highlight not only the challenges inherent in representing complex security datasets but also the innovative methods and tools that have been developed to confront these challenges. This body of work serves as both inspiration and a technical benchmark for our project, guiding our efforts in harnessing visualization techniques to unravel and elucidate the complexities of network security data.

III. Dataset

A. Datasets Description

*1) **KDD Cup 1999 Dataset:*** The KDD Cup 1999 dataset is a classic in the field of network intrusion detection. It is available on UCI's Machine Learning Repository.The dataset offers around 4.9 million records, each with 41 features, which include basic features of individual TCP connections, content features within a connection, and traffic features computed over a window interval.Intrusions are classified into four main categories: DOS, R2L, U2R , and Probing. The target feature is named as 'label'.Some of the features include 'duration', 'protocol-type', 'service', 'flag', 'src-bytes','dst-bytes'.



```
   duration protocol_type  ... dst_host_srv_rerror_rate    label
0         0           tcp  ...                      0.0  normal.
1         0           tcp  ...                      0.0  normal.
2         0           tcp  ...                      0.0  normal.
3         0           tcp  ...                      0.0  normal.
4         0           tcp  ...                      0.0  normal.
```

Fig. 1. First 5 rows of KDD Cup 1999 Dataset.

Attack types are the target variables present in the KDD-CUP 1999 dataset with feature name as 'label'. Fig.2 shows the various types of attacks in the Dataset.

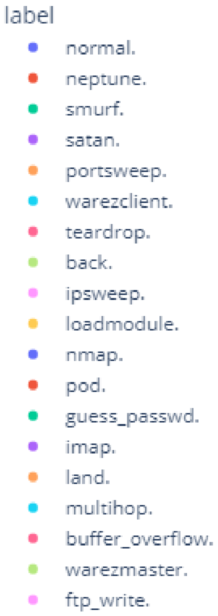
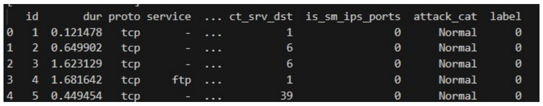


Fig. 2. Attack Types of KDD Cup 1999 Dataset.

2) ***UNSW-NB15 Dataset:*** The UNSW-NB15 dataset includes a comprehensive range of modern attack types.The dataset features over 2.5 million records with 49 features, encompassing a more diverse and contemporary set of features. some of the features include 'id' , 'label' , 'ct-dst-sport-ltm', 'dinpkt', 'tcprtt', 'attack-cat'



```
   id       dur proto service  ... ct_srv_dst  is_sm_ips_ports  attack_cat  label
0   1  0.121478   tcp       -  ...          1                0      Normal      0
1   2  0.649902   tcp       -  ...          6                0      Normal      0
2   3  1.623129   tcp       -  ...          6                0      Normal      0
3   4  1.681642   tcp     ftp  ...          1                0      Normal      0
4   5  0.449454   tcp       -  ...         39                0      Normal      0
```

Fig. 3. First 5 rows of UNSW-NB15 dataset.

Attack types are the target variables present in the UNSW-NB15 Dataset. Fig.4 shows the various types of attacks in the Dataset.The attacks are classified into nine categories, including Exploits, Fuzzers, Generic, Reconnaissance, DoS, Analysis, Backdoors, Shellcode,Normal and Worms.

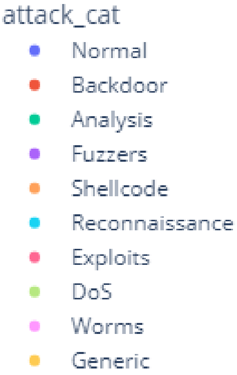


Fig. 4. Attack Types of UNSW-NB15 dataset.

B. Dataset Preprocessing

Data preprocessing included Loading the KDD Cup 1999 and UNSW-NB15 datasets into pandas DataFrames and then identifying and transforming categorical features (like 'proto', 'service', 'state' for UNSW-NB15, and 'protocol_type', 'service', 'flag' for KDD Cup) into numerical formats using Label Encoding. Continuted with Employing a Random Forest Classifier to determine the importance of features in both datasets.Further split the dataset into training and testing sets using train_test_split for model validation.Finally,Applied different scaling techniques (like Min-Max Scaling, Quantile Transformer) to standardize the feature ranges in both datasets.

C. Incorporated Visualization Techniques

The Visualization Techniques employed in the project for two datasets aim to identify class overlap and class imbalance problems. The techniques that are used in the project are explained below:

1) ***Nearest Shrunken Centroid (NSC):*** The NSC method is incorporated to determine the central vectors of each class within the datasets.

2) ***Mahalanobis Distances:*** Mahalanobis Distance is employed to measure the separation between class centroids, as calculated by NSC. This metric aids in identifying how distinct the classes are from one another in the feature space, a crucial step for outlier detection in intrusion detection systems.

3) ***PCA (Principal Component Analysis):*** PCA is leveraged to reduce the high-dimensional cybersecurity datasets while preserving essential variance. This simplification is critical for transforming complex data into an interpretable format that maintains the integrity of the original information content.

*4) **t-SNE (t-Distributed Stochastic Neighbor Embedding)**:* Through t-SNE, high-dimensional data is effectively mapped into a lower-dimensional space, which uncovers clusters and patterns. This visualization is particularly adept at revealing the nuanced groupings within our datasets, enhancing the pattern recognition capabilities.

*5) **K-Means Intercluster Distance Map**:* By applying K-Means clustering, inter-cluster distances can be visualized that reflect the diversity and similarity of data points within the datasets. This map is integral to understanding the structure of clusters formed in the context of network security analysis.

*6) **UMAP (Uniform Manifold Approximation and Projection)**:* Similar to t-SNE, UMAP provides dimensionality reduction but with faster computation and often more meaningful representations, particularly in capturing the global structure of data.

*7) **Parallel Coordinates**:* This multi-dimensional visualization technique is pivotal in the project for inspecting all features and their relationships simultaneously. The parallel coordinates plot is invaluable for comprehending the intricate relationships between attributes and pinpointing outliers and trends across the datasets.

D. *Incorporated Packages*

Each of these packages plays a crucial role in the data processing, analysis, modeling, and visualization pipeline, and they are widely used. The Incoporated packages list are given below Here's a brief overview of each package and its primary use:

*1) **Pandas**:* A foundational package for data manipulation and analysis. It provides data structures like DataFrames, making it easier to manipulate and analyze structured data.

*2) **Scikit-learn**:* A versatile machine learning library. It includes various tools for data preprocessing, model building, and evaluation.

*3) **Plotly**:* A graphing library that makes interactive, publication-quality graphs online. It includes ‘px‘ (Plotly Express) for simple syntax, and ‘go‘ (Graph Objects) for more customizable plots.

*4) **TSNE (‘sklearn.manifold.TSNE‘)**:* T-distributed Stochastic Neighbor Embedding, a tool for visualizing high-dimensional data by reducing it to two or three dimensions.

*5) **Matplotlib**:* A plotting library for creating static, interactive, and animated visualizations in Python.

*6) **UMAP**:* Uniform Manifold Approximation and Projection, another dimension reduction technique, particularly useful for visualization.

*7) **Dash**:* A framework for building interactive web applications. It's particularly suited for creating data visualization interfaces. dash, dcc ,html, and dash bootstrap components are parts of this framework.

IV. Methodology and Implementation

The groundwork for a comprehensive analysis of cybersecurity data is set by the methodology, with the aim of developing robust models for intrusion detection. The iterative process includes data standardization during preprocessing, focusing on relevant information through feature selection, understanding data characteristics through visualization, and addressing issues such as class overlap and imbalance that could impact the classification performance.

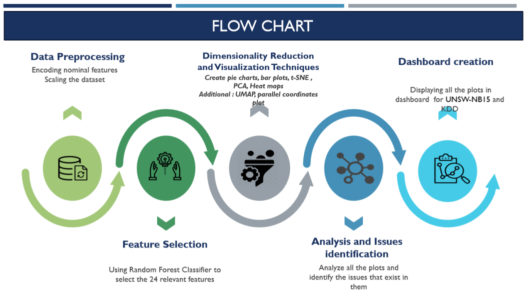

FLOW CHART
Data Preprocessing
Encoding nominal features
Scaling the dataset
Dimensionality Reduction and Visualization Techniques
Create pie charts, bar plots, t-SNE , PCA, Heat maps
Additional : UMAP, parallel coordinates plot
Dashboard creation
Displaying all the plots in dashboard for UNSW-NB15 and KDD
Feature Selection
Using Random Forest Classifier to select the 24 relevant features
Analysis and Issues identification
Analyze all the plots and identify the issues that exist in them

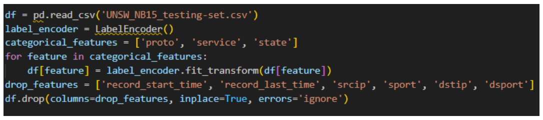
Fig. 5. Flow Chart

A. *Data Preprocessing*

The datasets for the project were obtained from their respective sources, namely the KDD Cup 1999 and UNSW-NB15 datasets. The initial step in the data preprocessing phase, a critical stage in dataset preparation for analysis, involved employing the same implementation for both datasets.

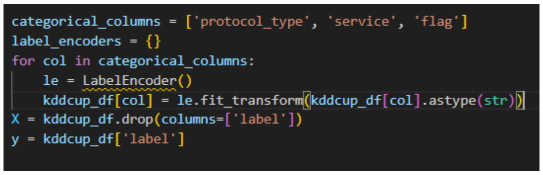
Data quality assessment, which encompassed evaluating aspects such as missing values and categorical data types, was conducted as part of the preliminary analysis. Subsequently, Label Encoding was applied to convert categorical features into a machine-readable format.

To normalize the distribution of feature values, Feature Scaling was implemented using StandardScaler. This step was taken to ensure consistency in the treatment of both datasets.



```
df = pd.read_csv('UNSW_NB15_testing-set.csv')
label_encoder = LabelEncoder()
categorical_features = ['proto', 'service', 'state']
for feature in categorical_features:
    df[feature] = label_encoder.fit_transform(df[feature])
drop_features = ['record_start_time', 'record_last_time', 'srcip', 'sport', 'dstip', 'dsport']
df.drop(columns=drop_features, inplace=True, errors='ignore')
```

Fig. 6. Preprocessing of UNSW-NB15 Dataset.



```
categorical_columns = ['protocol_type', 'service', 'flag']
label_encoders = {}
for col in categorical_columns:
    le = LabelEncoder()
    kddcup_df[col] = le.fit_transform(kddcup_df[col].astype(str))
X = kddcup_df.drop(columns=['label'])
y = kddcup_df['label']
```

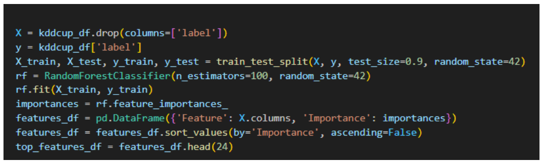
Fig. 7. Preprocessing of KDD Cup 1999 Dataset.

*1) **KDD Cup 1999 Dataset**:* The dataset is loaded without headers, and column names are assigned according to the documentation of the dataset. Redundant features are eliminated to enhance model performance. Nominal features like protocol types, services, and flags are transformed into numerical values through label encoding. Subsequently, the data is partitioned into features (X) and target (y), where the target corresponds to the ’label’ column. The features undergo scaling to normalize the distribution, ensuring that the model is not biased by the range of feature values.

*2) **UNSW-NB15 Dataset**:* Similar preprocessing steps are performed as those for the KDD Cup 1999 dataset. Moreover, extraneous features that may not contribute to the predictive model, such as source and destination IP addresses, are excluded. The dataset is partitioned into training and testing sets, with the preservation of class distribution through stratified sampling.

B. Feature Selection

Feature selection is performed to identify the most informative features that contribute to accurate classification. The evaluation of feature importance was conducted using the RandomForestClassifier. Subsequently, a subset of the top 24 features was selected from both the KDD Cup 1999 and UNSW-NB15 datasets for further analysis based on the importance scores. The identical process was employed for selecting the relevant features from both datasets.

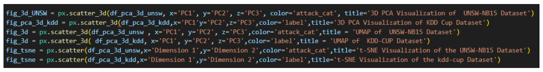


```
X = kddcup_df.drop(columns=['label'])
y = kddcup_df['label']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.9, random_state=42)
rf = RandomForestClassifier(n_estimators=100, random_state=42)
rf.fit(X_train, y_train)
importances = rf.feature_importances_
features_df = pd.DataFrame({'Feature': X.columns, 'Importance': importances})
features_df = features_df.sort_values(by='Importance', ascending=False)
top_features_df = features_df.head(24)
```

Fig. 8. Feature Selection of KDD Cup 1999 Dataset.

C. Exploratory Data Analysis and Visualization

Data visualization techniques are employed to explore data and extract insights regarding the inherent structure and relationships. Techniques such as PCA, t-SNE, and UMAP are utilized for the purpose of dimensionality reduction to facilitate visualization. Various types of plots, including pie charts, bar plots, 3D scatter plots for PCA, heatmaps for Mahalanobis distances, parallel coordinates plots, and intercluster distance maps using KMeans clustering, are generated to visualize the data. These plots aid in identifying issues within the datasets. The code for applying these visualizations to the datasets is provided in the figure.



```
fig_3d_UNSW = px.scatter_3d(df_pca_3d_unsw, x='PC1', y='PC2', z='PC3', color='attack_cat', title='3D PCA Visualization of  UNSW-NB15 Dataset')
fig_pca_3d_kdd = px.scatter_3d(df_pca_3d_kdd,x='PC1'y='PC2',z='PC3',color='label',title='3D PCA Visualization of KDD Cup Dataset')
fig_3d = px.scatter_3d(df_pca_3d_unsw , x='PC1', y='PC2', z='PC3',color='attack_cat',title = 'UMAP of  UNSW-NB15 Dataset')
fig_3d = px.scatter_3d( df_pca_3d_kdd, x='PC1', y='PC2', z='PC3',color='label',title = 'UMAP of  KDD-CUP Dataset')
fig_tsne = px.scatter(df_pca_3d_unsw,x='Dimension 1',y='Dimension 2',color='attack_cat',title='t-SNE Visualization of the UNSW-NB15 Dataset')
fig_tsne = px.scatter(df_pca_3d_kdd,x='Dimension 1',y='Dimension 2',color='label',title='t-SNE Visualization of the kdd-cup Dataset')
```

Fig. 9. Visualizations of KDD Cup 1999 Dataset and UNSW-NB15 Dataset.

D. Analysis and Issues Identification

After the plots have been obtained, an analysis reveals two visualization technique problems in the two datasets, namely, class overlap and class imbalance. The class imbalance problem can be detected through barplots, PCA, and k-means, while the class overlap problem can be identified through t-SNE and k-means, where the overlap is indicated by circles highlighting the overlap of all classes.

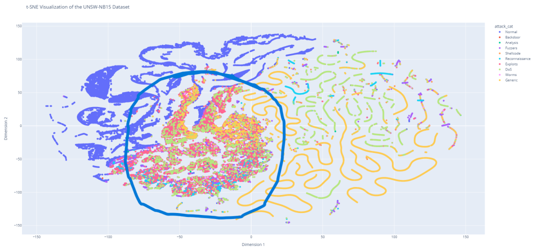
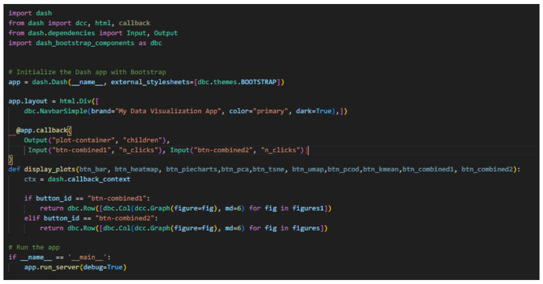


Fig. 10. Visualization of class overlap

E. DashBoard Creation

A dashboard has been developed using Plotly and Dash to provide interactive visualizations for both the KDD Cup 1999 dataset and the UNSW-NB 15 dataset. Two buttons are incorporated on the web page, each corresponding to one dataset. When a button is clicked, all the associated plots for that specific dataset will be presented.



```
import dash
from dash import dcc, html, callback
from dash.dependencies import Input, Output
import dash_bootstrap_components as dbc


# Initialize the Dash app with Bootstrap
app = dash.Dash(__name__, external_stylesheets=[dbc.themes.BOOTSTRAP])

app.layout = html.Div([
    dbc.NavbarSimple(brand="My Data Visualization App", color="primary", dark=True),])

  @app.callback(
    Output("plot-container", "children"),
     Input("btn-combined1", "n_clicks"), Input("btn-combined2", "n_clicks")]
)
def display_plots(btn_bar, btn_heatmap, btn_piecharts,btn_pca,btn_tsne, btn_umap,btn_pcod,btn_kmean,btn_combined1, btn_combined2):
    ctx = dash.callback_context

    if button_id == "btn-combined1":
        return dbc.Row([dbc.Col(dcc.Graph(figure=fig), md=6) for fig in figures1])
    elif button_id == "btn-combined2":
        return dbc.Row([dbc.Col(dcc.Graph(figure=fig), md=6) for fig in figures])

# Run the app
if __name__ == '__main__':
    app.run_server(debug=True)
```

Fig. 11. DashBoard Creation Code.

F. Teammember’s Contributions

*1) **SAI SRI CHARISHMA VADDI: KDD Cup 1999 Dataset Analysis**:*

- Data Loading and Preprocessing for KDD Cup 1999 dataset
- Feature Selection on KDD Cup 1999 dataset
- Advanced Analysis and Visualization for KDD Cup 1999 Dataset
- Implementation of PCA, t-SNE, and UMAP for dimensionality reduction and data visualization.
- Creation of parallel coordinates plot for selected features.
- Performing KMeans clustering and visualize clusters.
- Dashboard Development for KDD Cup 1999 Dataset
- Developing interactive plots (pie charts, bar plots, heatmaps, etc.) for the KDD Cup 1999 dataset using Dash and Plotly.
- Integratation of these plots into the web-based dashboard.
- Documentation of report

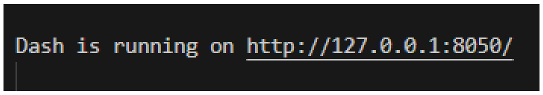
*2) **SAI SREEJA SINGIREDDY: UNSW-NB15 Dataset Analysis**:*

- Data Loading and Preprocessing for UNSW-NB15 Dataset
- Feature Selection on UNSW-NB15 Datset

- Advanced Analysis and Visualization for UNSW-NB15
- Implementation of PCA, t-SNE, and UMAP for dimensionality reduction and data visualization.
- Creation of parallel coordinates plot for selected features.
- Performing KMeans clustering and visualize clusters.
- Dashboard Development for UNSW-NB15 Dataset
- Developing interactive plots (pie charts, bar plots, heatmaps, etc.) for the UNSW-NB15 dataset using Dash and Plotly.
- Integration of these plots into the web-based dashboard.
- Documentation of report

V. DashBoard creation

A web page has been designed to display a collective set of plots in the form of a dashboard using the Dash module in Python. The Dash app is initiated with the command app.run_server(debug=True), which starts the web server and enables live reloading for development purposes. The application is hosted on the server, and upon clicking the provided link on the page, it redirects to another web page accessible at ”http://127.0.0.1:8050/”.



```
Dash is running on http://127.0.0.1:8050/
```

Fig. 12. Webpage link generated.

The web page features a dashboard UI, as depicted in the figure below. There are six buttons available for generating various visualizations, including heatmap plots, pie charts, bar plots, PCA plots, t-SNE plots, etc. Additionally, there are two more buttons labeled ”Dashboard for KDD Cup 1999 dataset” and ”Dashboard for UNSW-NB15 dataset.”

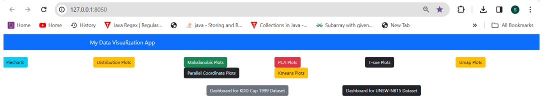


Fig. 13. Homepage showing all the buttons.

After clicking on each plot, a comparison of six buttons is displayed, similar to the one shown in the figure. When the ”barplots” button is clicked, a figure appears containing two barplots, with each plot representing a different dataset. This approach allows for a visual comparison of the two datasets, facilitating clear analysis and interpretation.

Dashboard for KDD Cup 1999 Dataset
Dashboard for UNSW-NB15 Dataset

Fig. 14. Buttons to visualize all the individual plots for each dataset.

There are two buttons labeled ”Dashboard for UNSW-NB15 Dataset” and ”Dashboard for KDD-CUP 1999 Dataset.” Each button displays comprehensive plots generated for its respective dataset. Fig.15 and Fig.16 illustrates the dashboards for these datasets.

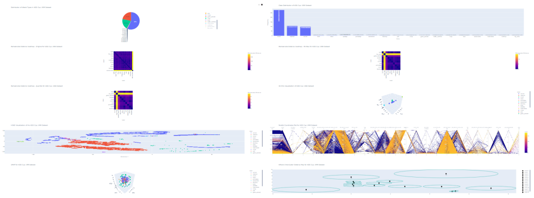


Fig. 15. KDD Cup 1999 Dataset Dashboard.

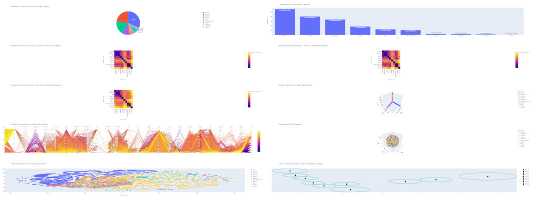


Fig. 16. UNSW-NB15 Dataset Dashboard

By using this dashboard, users can interact with the data in real-time, adjusting parameters, and focusing on specific areas of interest. This interactivity enhances the user’s ability to make informed decisions and derive meaningful conclusions from the data.

VI. Results and Analysis

A. Class Distribution of Datasets

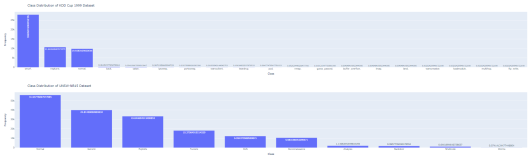


Fig. 17. Class Distribution of KDD Cup 1999 and UNSW-NB15 Dataset.

The bar graphs for the KDD Cup 1999 and UNSW-NB15 datasets reveal a stark class imbalance, with dominant classes like ’smurf’ and ’neptune’ in KDD Cup 1999, and ’Normal’ and ’Generic’ in UNSW-NB15 overshadowing minority classes. This imbalance risks classifier overfitting to more frequent classes and underdetecting less common but potentially critical attack types.

B. Pie Charts

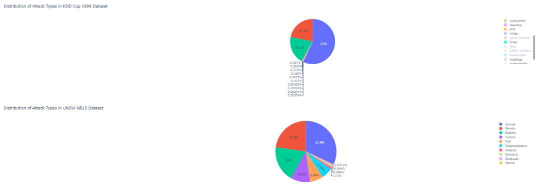


Fig. 18. Distribution of Attack Types in KDD Cup 1999 and UNSW-NB15 Dataset.

The pie charts further illustrate this disparity, quantifying the exact percentage share of each class, reinforcing the observations made from the bar graphs. The pie charts for the KDD Cup 1999 and UNSW-NB15 datasets reveal significant class imbalances, with a few classes dominating each dataset, which could impair the performance of classifiers.

C. *The Mahalanobis Distance of the centroids*

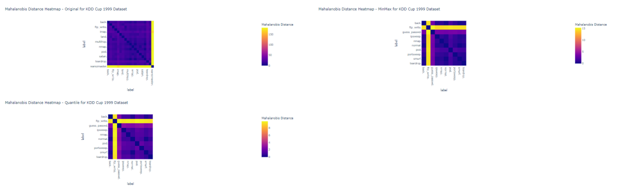


Fig. 19. Mahalanobis Distance before and after normalization for KDD Cup 1999 Dataset.

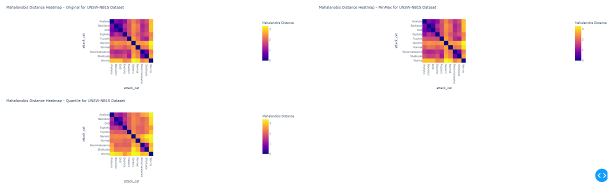


Fig. 20. Mahalanobis Distance before and after normalization for UNSW-NB15 Dataset.

The Mahalanobis distance heatmaps for the KDD Cup 1999 and UNSW-NB15 datasets reveal that normalization techniques like Quantile Transfomer and Min-Max Scaler clarify class separability. Pre-normalization heatmaps display significant class overlap, particularly for minority classes, which can degrade classifier performance. Post-normalization, the increased distinction between classes suggests a reduction in overlap, offering a potential boost to classification accuracy and the effectiveness of intrusion detection models.

D. *PCA Visualizations*

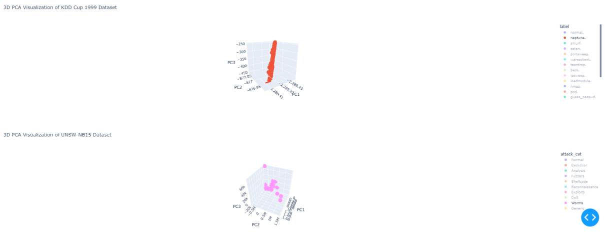


Fig. 21. Visualization of the individual classes using PCA for Within-class imbalance

The PCA visualizations reveal within-class imbalance for individual classes within the UNSW-NB15 and KDD Cup 1999 datasets, with the ’Worms’ class in UNSW-NB15 showing a tight cluster indicative of a minority class, and the ’normal’ class in KDD Cup 1999 exhibiting a wide spread, characteristic of a majority class.

The PCA visualizations for pairs of classes from the UNSW-NB15 and KDD Cup 1999 dataset display significant insights into class overlap and within-class imbalance. In the visualization, we can see distinct clustering for each class, which suggests a high class overlap probability. This highlights the challenges in distinguishing between these two classes, potentially leading to misclassifications in a predictive model.

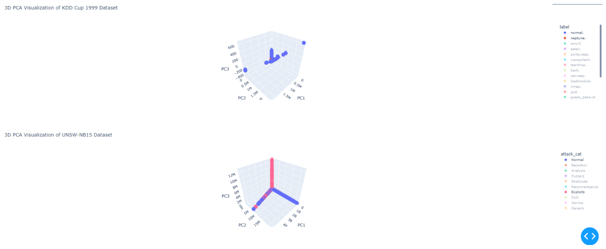


Fig. 22. Visualization of a pair of classes using PCA for Class Overlap

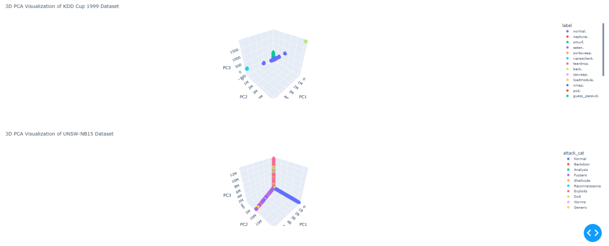


Fig. 23. Visualization of the entire dataset using PCA for Class Overlap

The PCA visualizations for all classes for the KDD Cup 1999 and UNSW-NB15 datasets illustrate the concept of class overlap, where the proximity of data points between different classes can lead to classification challenges. In the context of the KDD Cup 1999 dataset, the visualization shows classes with high overlap probability, suggesting that some attacks have similar patterns to normal traffic, which could result in a classifier confusing one for the other. For the UNSW-NB15 dataset, there’s a visible distinction between some of the attack classes and normal traffic, yet a potential for overlap still exists.

E. *t-SNE Visualizations*

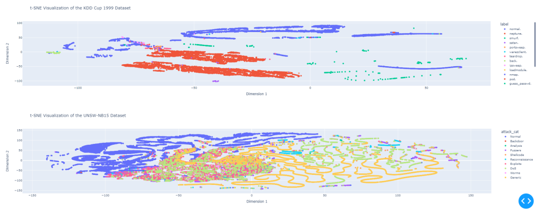


Fig. 24. Visualization of all classes using T-sne

The t-SNE visualization of the UNSW-NB15 and KDD Cup 1999 datasets reveals that classes are not distinctly separated but have multiple clusters of varying sizes spread across the two-dimensional space, indicating complex class structures with both large and small clusters within classes. There is significant overlap between clusters of different classes, complicating the task of classification and detection of malicious traffic. The analysis indicates a notable ”overlap problem” where attack classes mimic normal behavior, posing challenges for intrusion detection systems trained on this data.

F. *Inter-cluster distances and K-means Visualizations*

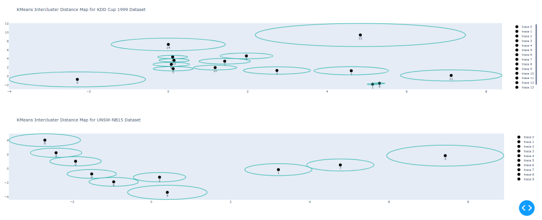


Fig. 25. Visualization of all classes using K-Means

The K-Means intercluster distance maps for the KDD Cup 1999 and UNSW-NB15 datasets reveal distinct clustering, with KDD Cup 1999 showing clearer separation, suggesting easier classification. The UNSW-NB15 dataset, however, shows clusters in closer proximity, indicating potential class overlap that could challenge classifier accuracy.

G. *UMap Visualizations*

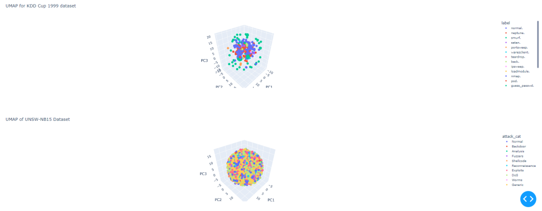


Fig. 26. Visualization of all classes using UMAP

UMAP visualizations of the KDD Cup 1999 and UNSW-NB15 datasets illustrate class distribution and interaction, indicating imbalances with densely populated clusters for some classes and sparse ones for others. Overlapping clusters signal class inseparability in feature space, posing classification challenges and potential misclassification risks.

H. *Parallel Coordinates Visualizations*

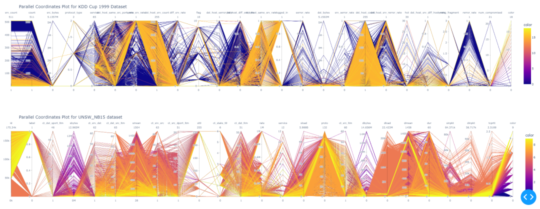


Fig. 27. Visualization of all classes using Parallel Coordinates

The parallel coordinates plots for the KDD Cup 1999 and UNSW-NB15 datasets reveal complex interactions between features across classes, with dense overlaps indicating potential class confusion and sparse areas suggesting unique class traits.

VII. CHALLENGES

Each technique, while shedding light on certain aspects, also unveiled new challenges.

A. *Imbalanced Dataset:*

The distribution of the classes in the datasets is often imbalanced, with a significant proportion of the data belonging to one class.

B. *Large and Complex Dataset:*

The dataset is large and complex. Processing and analyzing such a large dataset can be computationally intensive, requiring significant hardware resources

C. *Lack of Expertise:*

Developing accurate visualisations for datasets often requires expertise. Without proper domain expertise, it can be challenging to understand the nuances of the datasets and develop accurate results.

VIII. CONCLUSION

The challenges of class imbalance and class overlap in the UNSW-NB15 and KDD Cup 1999 datasets are addressed, both of which are pivotal for developing reliable intrusion detection systems.It was observed that while the KDD Cup 1999 dataset exhibits less pronounced class imbalance and overlap compared to UNSW-NB15, both datasets present unique complexities that necessitate meticulous preprocessing and visualization. We have performed several advanced visualization techniques like PCA, t-SNE, U-Map, Parallel Coordinates and K-means,on the two datasets.The findings from our visual analysis directly informed the development of our visualization dashboard, a tool designed to be both interactive and comprehensive, enabling users to effectively navigate and interpret these intricate datasets. This dashboard emerged as a critical asset in the project, facilitating an in-depth exploration of network traffic data.

IX. FUTURE WORK

The existing techniques, while effective, have room for enhancement, particularly in refining outlier detection and class representation. A plan to integrate advanced outlier detection algorithms like isolation forests, DBSCAN, and autoencoders to improve the accuracy of centroid calculations, a crucial aspect that was less precise in the current approach can be made.Additionally, new centroid calculation methods, such as density-based centroids, to better capture the true distribution of data, addressing the class overlap issue more effectively can be explored. Furthermore, an upgrade of the visualization dashboard can also be planned.Future enhancements to the visualization dashboard can include the integration of advanced analytical features like predictive analytics and customizable widgets. Enhanced interactivity can be achieved through the implementation of drill-down capabilities and data linking, while user experience can be optimized with customizable layouts and accessibility options.

References

[1] Lee HK, Kim SB. An overlap-sensitive margin classifier for imbalanced and overlapping data. Exp Syst Appl. 2018;98:72-83.

[2] Das S, Datta S, Chaudhuri BB. Handling data irregularities in classification: foundations, trends, and future challenges. Pattern Recogn. 2018;81:674-693.

[3] Kumar V, Sinha D, Das AK, Pandey SC, Goswami RT. An integrated rule based intrusion detection system: analysis on UNSW-NB15 data set and the real time online dataset. Clust Comput. 2020;23:1397-1418.

[4] Kanimozhi V, Jacob P. UNSW-NB15 Dataset Feature Selection and Network Intrusion Detection using Deep Learning

[5] Pezzotti, N., Lelieveldt, B. P., van der Maaten, L., Höllt, T., Eisemann, E., Vilanova, A. (2016). Approximated and user steerable tSNE for progressive visual analytics. IEEE transactions on visualization and computer graphics, 23(7), 1739-1752.

[6] Ruan, Z., Miao, Y., Pan, L., Patterson, N., Zhang, J. (2017). Visualization of big data security: a case study on the KDD99 cup data set. Digital Communications and Networks, 3(4), 250-259.

[7] Stahnke, J., Dörk, M., Müller, B., Thom, A. (2015). Probing projections: Interaction techniques for interpreting arrangements and errors of dimensionality reductions. IEEE transactions on visualization and computer graphics, 22(1), 629-638.

[8] M. Shyu, S. Chen, K. Sarinnapakorn, and L. Chang, “A novel anomaly detection scheme based on principal component classifier,”Proceedings of the IEEE Foundations and New Directions of Data Mining Workshop, in conjunction with the Third IEEE Intern

[9] https://onlinelibrary.wiley.com/doi/pdf/10.1002/spy2.331

[10] https://arxiv.org/ftp/arxiv/papers/2101/2101.05067.pdf

[11] https://research.unsw.edu.au/projects/unsw-nb15-dataset