Network Intrusion Detection

Piotr Bonar[a], **Maddox Hurlbert**[b] **and Sonia Serra**[c]

[a] *1759684*
[b] *1729809*
[c] *1753004*

Abstract—With the increasing sophistication of cyber threats and complexity of network traffic, this project explores machine learning techniques to enhance real-time network intrusion detection. Using the comprehensive UNSW-NB15 dataset, we apply feature reduction techniques such as PCA, clustering with K-Means, classification with KNN, and collaborative filtering for predictive threat modeling. PCA effectively reduced data dimensionality while preserving variance, improving training efficiency. KNN achieved high accuracy in classifying malicious and benign traffic, while collaborative filtering offered novel insights for attack prediction. Although each method has limitations, their combination demonstrated robust detection capabilities and future potential for cybersecurity applications.

Keywords—*Network Intrusion Detection, UNSW-NB15, Machine Learning, PCA, K-Means, KNN, Collaborative Filtering, Cybersecurity*

Contents

1. Introduction

With the ongoing escalation of cyber threats and the increasing intricacy of networked systems, detecting security breaches in real time has become not just a technical necessity but a strategic priority across both personal and organizational domains. Modern networks handle immense volumes of heterogeneous data, creating challenges for traditional detection methods, which often struggle to keep up with stealthy, low-footprint attacks and evolving threat vectors.

This growing demand for proactive and intelligent security solutions has driven the development of machine learning-based intrusion detection systems (IDS), which can analyze traffic patterns, learn from past attacks, and identify anomalies indicative of malicious behavior. By leveraging large-scale labeled datasets, these systems offer the potential to move beyond reactive defense toward predictive and adaptive security mechanisms.

In this project, we aim to explore such techniques by applying a combination of unsupervised and supervised learning algorithms to the UNSW-NB15 dataset — a modern and comprehensive benchmark for network intrusion detection research. Our goal is to understand how machine learning models can uncover attack signatures embedded in network traffic, reduce false positives, and ultimately assist in designing scalable solutions that can generalize across various types of cyber threats in real-world conditions.

2. About the Dataset

The University of New South Wales (UNSW) created a dataset, with many different data points and indicators for possible and real attacks, which we have decided to use to develop an effective way to search networks for intrusions and detecting the correlations between the attacks themselves and the networks properties by applying machine learning techniques to simplify and increase the accuracy of the detection machines.

The dataset, developed by the Australian Centre for Cyber Security at the UNSW, addresses more than previous representations as it is a much more comprehensive representation of modern network traffic and contemporary low-footprint attacks. Containing over 2.5 million records, each being described by 49 features (extracted using network security tools) that capture various traffic characteristics within the data, such as their IP addresses, ports, protocols, packet counts, flow statistics and more. Two network monitoring tools, Argus and Zeek, were used to track network connections, flow metrics, and deep protocol analysis to detect the network's behavioral anomalies and outside threats. Each recorded interaction is either labeled as normal (no attack) or one of nine attack categories: Fuzzers, Analysis, Backdoors, DoS, Exploits, Generic, Reconnaissance, Shellcode, and Worms. The dataset is split into two sets, a training set with 175,341 records, which we will primarily be using, as well as a testing set of 82,332 records, which we will try to use to understand the intrusion giveaways. It offers a very diverse set of normal traffic and various types of security breaches of networks, which will allow us to gather a deeper understanding of the machine learning models we will be using to understand the dataset.

3. Data Preparation and Feature Reduction

To begin our research we separated our data into two sets, training and testing, which would respectively be eventually used to train a model, but we will just be working with the training set, to develop a successful way to analyze and predict future data. We also ensured a reliable model using nearly 175,000 samples, two-thirds of the samples with attacks, while the rest unattacked. With 45 labels, we had already eliminated those of which contain no relevant info for training like label, attack category, and Id number, that we had read of each data point, we were tasked with reducing the features to ensure smoother future training sessions. To start the reduction process, we began with a correlation matrix, to compute which of our factors help similar data to others. Perhaps we could combine them using analysis of the data's principal components.

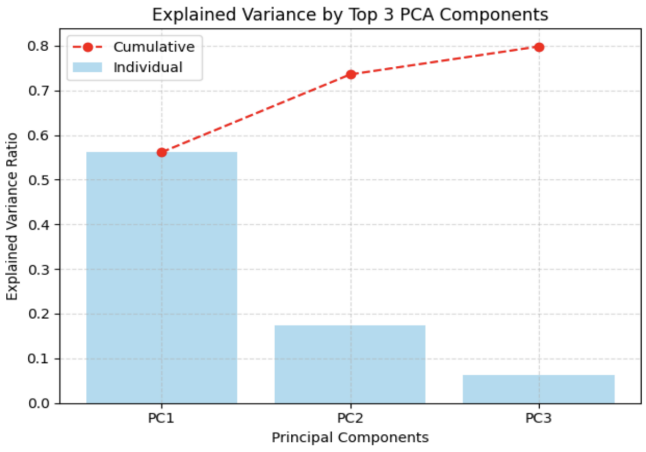


Figure 1. Explained variance ratio for the top 3 principal components.

With 15 feature similarities, we definitely would be able to condense the dataset features. We remodeled one feature of each similar set and then created another correlation matrix between our new features and the attack categories. Using principal component anal-

ysis, a linear dimensionality reduction technique with applications in exploratory data analysis, visualizations and data preprocessing. After standardizing our data using standard scalar, we then used K-means with 3 clusters on the Component analysis of two components. Observing the PCA and K-means clustering we found 3 nearly placed clusters, meaning that they could possibly not generalize well with new data, so we applied the min max scaler, and retried PCA which had shown us more than 70% of the variance explained, as opposed to the near 55% of the first analysis. In the graphical representation of the second application, we see a much more defined set of clusters, between those attacked and unattacked. Then we applied PCA with three components which resulted in yet another increase up to nearly 80% variance explained.

Analyzing these features, we found the most prominent ones for each principal component:

Principal Component 1 (PC1):

- `dwin`
- `dttl`
- `stcpb`
- `dtcpb`
- `sttl`

Principal Component 2 (PC2):

- `sttl`
- `dttl`
- `dmean`
- `ct_state_ttl`
- `dload`

Principal Component 3 (PC3):

- `ct_srv_dst`
- `ct_src_dport_ltm`
- `ct_src_ltm`
- `ct_dst_sport_ltm`
- `stcpb`

Figure 2. 2D PCA visualization of the UNSW-NB15 dataset colored by attack category. The projection onto the first two principal components (PC1 and PC2) reveals distinct clusters corresponding to different types of network activity, helping distinguish between normal traffic and various types of cyberattacks.

4. Clustering and Classification

After applying PCA, we observed that K-means was not the most suitable algorithm for our classification objective, as the dataset was already labeled to distinguish between benign and malicious traffic. K-means is primarily designed for unsupervised learning, where labels are absent and clusters are formed based on similarity rather than predefined categories. However, its application in this project proved valuable in revealing the underlying structure of the data and served as a useful preprocessing step for noise reduction prior to supervised learning. It also contributed later during the clustering phase of the recommender system.

We computed the inertia values for different cluster sizes and found that K values between 3 and 5 yielded the most optimal results. Visual analysis confirmed that $K = 3$ provided the best separation between normal and attack traffic. Subsequently, we evaluated classification performance using KNN on the PCA-reduced data. Once again, the best performance was achieved at $K = 3$, with an overall accuracy of 93%. Further evaluation of the classification metrics showed high precision and recall values for both traffic categories. The results are summarized in Table 1.

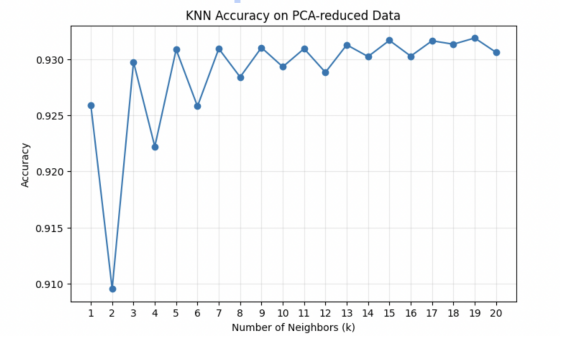


Figure 3. KNN Accuracy across varying values of k.

Category	Accuracy	Precision	Recall
Normal Traffic	93%	94%	>80%
Attack Traffic	93%	92%	>95%

Table 1. Classification performance on PCA-reduced data using $K = 3$ with the KNN algorithm. The model demonstrates high accuracy and strong differentiation between normal and malicious traffic.

5. Collaborative Filtering

Then our goal to analyze the data was to consider whether or not a collaborative filtering-based recommender system would be able to provide meaningful insight in regards to predicting potential threats based on the historical data we had to work with. We sought after finding which attributes of the traffic were present during each type of attack to gain insight as to what allowed the breach. Maybe certain types of services are more prone to certain types of attacks. Comparing the types of attacks to the types of services they attacked. Of the 119,341 attacked data points, 57,656 of those were unrecognized services, 48.31% of the attacks.

We then created an interaction matrix which visualized the associations between each attack and the service provided, ignoring those of which were undeclared, with most of the attacks being 39,116 generic attacks on DNS. Then we compared the similarities between the features that were significant to the attacks with the other significant features of other attacks, or more simply, how similar the features were that allowed the attacks.

Comparing the attacks of analysis, backdoor and DoS to the attacks of fuzzers, shellcode, reconnaissance, and worms were the least similar to each other, ranging between 0.31 and 0.2, whereas generic compared with the three Analysis, Backdoor, and DoS were nearly completely different with a similarity of less than 0.1 in all categories while also including Exploits. We then wanted to find which attacks were most similar to each attack, so we found the top 3 similar attacks to each and found out that the attacks all had very similar features.

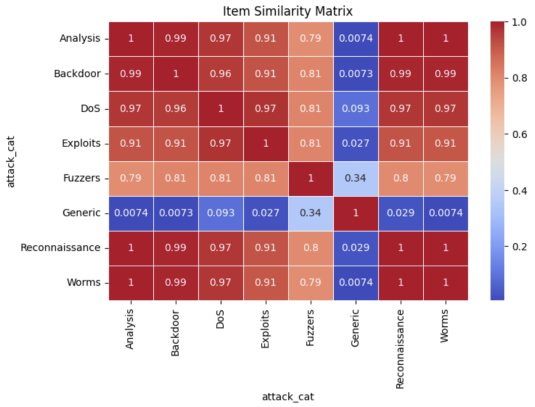
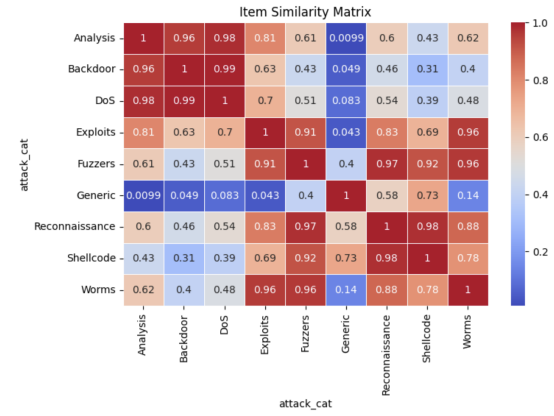


Figure 4. Similarity matrix between attack categories based on feature profiles.



a d

Figure 5. Similarity matrix between attack categories based on feature profiles.

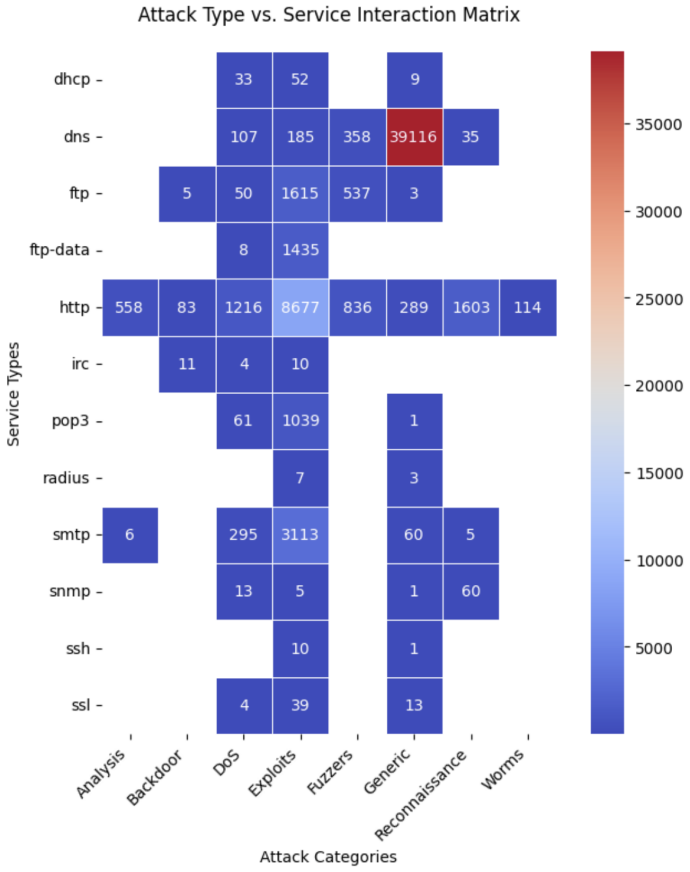


Figure 6. Interaction matrix between attack categories and targeted network services. Each cell indicates the number of occurrences of a specific attack type against a given service.

The features read for backdoor were 0.994 similar to those of DoS and 0.957 similar to those of Analysis attacks and Reconnaissance with 0.974 similarity to Fuzzers.

6. Prediction and Evaluation

Then we wanted to predict the most likely attack on the traffic that our dataset classified as normal, so as to test our system, and then classify the attack. Since there were so many samples of which were not attacked we extracted a smaller random sample of 5000 users to perform the test on. To test, we took a random sample and found its five nearest neighbors to assign the possible attack to it. As for ID number 6256, the 5 nearest neighbors were all fuzzers, so we were able to agree that it would be the most likely contender if the site was to be attacked. This algorithm would help cyber-security companies to specialize their security systems for certain attacks and make them less prone to future breaches.

7. Conclusion

In summary, our application of PCA effectively reduced the feature dimensionality while preserving the key features and variance, improving our model's efficiency. The algorithm was able to simplify and speed up training as well, at a small cost of reducing interpretability, and the optimal number of components depending on the dataset, for example t-3 PCs explained around 80% of the total variance. K-means clustering revealed meaningful grouping of traffic patterns, with optimal clusters between 3 and 5. It grouped similar traffic patterns, but required a lot of manual labeling of the clusters as malicious or normal behavior. The KNN combined with the PCA reduction data achieved an accuracy of about 93%, which is impressive, but also comes with certain drawbacks such as time consumption during training. The system achieved a solid performance in detecting certain types of intrusions, demonstrating a good generalization if used for future research. The collaborative filtering system provided an innovative perspective by attempting to classify the most susceptible attack of certain traffic data based on the past data researched in our project. For example, it analyzed interactions between certain user IDs and attack categories, such as anticipating exploits following a DoS transaction. However its usefulness comes with caveats like requiring numerous past data of attack patterns to make accurate predictions and faces challenges with the cold-start problem when new or previously unseen threats emerge. But overall, the combined techniques used throughout the research period offered many complementary insights into network traffic behavior, enhancing the robustness of the intrusion system.

As for the UNSW-NB15 dataset itself, each technique that we applied has very distinct advantages and disadvantages. PCA excelled at dimensional reduction, but reduced the interpretability of individual features. K-means was valuable for unsupervised detection, but required a lot of post-processing to label each cluster and is very sensitive to outside noise. KNN achieved a high accuracy of 93% and was very straightforward during implementation, but became very slow and resource-intensive with such a large dataset as we had, which is difficult to keep effective. The Collaborative filtering revealed hidden attack patterns and suggested potential future threats, yet it loses effectiveness when the attack interactions become sparse. Overall, the combination of PCA and KNN was the most successful in our research for the university, balancing accuracy and training speed, while collaborative filtering provided a lot of info during our exploration of the dataset.

For future work, our research could offer brief insight as it was primarily used for this specific dataset and was generally a simple approach, but if developed further, could offer a lot of useful information to cybersecurity companies and their systems to protect our users and businesses online from digital crime.