Enhancing Network Intrusion Detection through Data Dimensionality Reduction Using Classical and Deep Learning Approaches

Agnieszka Wosiak
Lodz University of Technology
Łódź, Poland *agnieszka.wosiak@p.lodz.pl*

Kacper Świąder
Lodz University of Technology
Łódź, Poland *250004@edu.p.lodz.pl*

Rafał Woźniak
Lodz University of Technology
Łódź, Poland *rafal.wozniak@p.lodz.pl*

Adam Niewiadomski
Lodz University of Technology
Łódź, Poland *adam.niewiadomski@p.lodz.pl*

Abstract

Network intrusion detection systems face high-dimensional traffic, which degrades accuracy and raises computational costs. We evaluate Principal Component Analysis (PCA), Recursive Feature Elimination (RFE), and a deep autoencoder on the UNSW-NB15 dataset using eight classifiers. RFE delivers peak accuracy (92.5%) with minimal variability. PCA restores near-baseline accuracy while preserving 95% variance with minor tuning. The autoencoder yields nonlinear embeddings but demands extensive training and trails classical methods. These findings guide the selection of reduction strategies under accuracy requirements and resource constraints.

Keywords: dimensionality reduction, deep learning, PCA, autoencoder, cybersecurity.

1. Introduction

The exponential growth of internet-connected devices and sophisticated cyber-attacks has made real-time network intrusion detection systems (NIDS) essential. Modern NIDS must process high-dimensional traffic, which inflates computational cost and can obscure critical attack signatures. This "curse of dimensionality" not only increases training and inference times but also degrades accuracy by introducing redundant or noisy attributes.

To address these challenges, classical reduction techniques such as Principal Component Analysis (PCA) and Recursive Feature Elimination (RFE) prune feature space while preserving key information [1]. Moreover, hybrid heuristics combine filter and wrapper strategies [12], and approaches like Gini Impurity–based Weighted Random Forest (GIWRF) balance reduction with minimal performance loss [2]. Deep autoencoders (DAEs) learn nonlinear embeddings that capture subtle attack patterns and can be made robust via denoising or contractive variants [11]. However, prior work typically evaluates these methods in isolation—using different datasets and classifiers—which hinders direct comparison of their relative merits.

This paper evaluates classical and deep learning–based dimensionality reduction methods across multiple classifiers, examining their effects on detection accuracy and computational cost. We measure how varying feature counts affect training and inference times and analyze key metrics to inform operational trade-offs. A unified pipeline demonstrates real-world speedups

in model retraining and packet classification without degrading detection quality. Finally, we provide practical guidelines for selecting reduction techniques that balance resource constraints with security requirements.

2. Materials and Methods

We evaluate classical and deep-learning–based dimensionality-reduction methods on a realistic intrusion-detection benchmark and assess their impact on multiple classifiers in terms of detection performance and computational cost.

Data Characteristics. The UNSW-NB15 dataset comprises 2,540,044 records described by 49 attributes representing benign and attack classes [4, 5, 6, 7], [9]. These records were split into a training set of 175,341 samples and a testing set of 82,332 samples, preserving a realistic distribution of normal and malicious flows. The original features were standardized to have a mean of zero and a variance of one to ensure comparability across metrics.

Data Preprocessing. To address the severe imbalance between benign and attack classes, rather than simply duplicating minority-class records—which can lead to overfitting—we applied the Synthetic Minority Oversampling Technique (SMOTE), generating synthetic minority-class examples along feature-space interpolations to balance the training data. As a result, our training set expanded from 175 341 to approximately 260 000 records, achieving an approximately 1:1 ratio between benign and malicious classes.

Data Dimensionality Reduction. We compare three methods for compressing the original 49-dimensional feature space into compact embeddings. PCA linearly projects standardized features onto orthogonal principal axes ranked by the explained variance [8], retaining those components that, in our approach, together account for 95% of the total variance. RFE fits a supervised algorithm to rank features by importance and then recursively removes the least informative variables until the target subset size is reached [10]. The choice of algorithm has a significant impact on the process's efficiency. For this study, we employed the Random Forest algorithm. A deep autoencoder is a neural network model whose primary function is to reconstruct the input data by compressing it into a more concise, lower-dimensional representation. DAE learns effective data representations by reducing dimensionality and extracting the most salient features and then attempts to reconstruct the original data from this compressed form [3]. Our autoencoder employs layer sizes 49 / 32 / 16 / 8 / 16 / 32 / 49 with ReLU activations and 20% dropout, trained for 100 epochs to minimize mean-squared-error loss, yielding an 8-dimensional nonlinear embedding.

Classification Methods. We evaluated eight classifiers chosen to span the major learning paradigms in intrusion detection: rule-based (Decision Tree), ensemble (Random Forest, Extremely Randomized Trees, Gradient-Boosted Trees, Adaptive Boosting), instance-based (k-Nearest Neighbors), kernel-based (Support Vector Machine), and neural (Multilayer Perceptron). Hyperparameters for each model were selected through preliminary grid searches with 5-fold cross-validation on the training set to balance bias and variance. The final settings were: Decision Tree using Gini impurity with no depth limit; Random Forest and Extra Trees with 100 estimators and $\sqrt{p}$ features per split; k-NN with k=5; SVM with an RBF kernel and C=1.0; GBT with 100 estimators and a learning rate of 0.1; AdaBoost with 50 estimators and learning rate=1.0; and MLP with one hidden layer of 100 ReLU neurons trained by Adam (learning rate=0.001, batch size=64). This consistent tuning approach ensures that observed performance differences stem from the dimensionality-reduction methods rather than model configuration.

3. Results and Discussion

The experiments used a uniform pipeline to isolate the impact of dimensionality reduction on classification. We first split UNSW-NB15 into a training set and a held-out test set, standardizing

all features and balancing the training data with SMOTE. We then applied PCA, RFE, and a deep autoencoder to reduce dimensionality to 8, 16, and 24 features (≈ 15%, 33%, 50% of the original). Finally, we trained eight classifiers—Decision Tree (DT), Random Forest (RF), k-NN, SVM, Gradient-Boosted Trees (GBT), AdaBoost, Extremely Randomized Trees (ET), and MLP—on both the full and each reduced feature set.

Table 1. Accuracy and F1-score ranges across dimensionality-reduction methods and classifiers.

				MinScore				MaxScore				MaxDiff		
Method	**T**	**S**	**SR[%]**	**ACC[%]**	**F1[%]**	**C**	**F**	**ACC[%]**	**F1[%]**	**C**	**F**	**ACC[%]**	**C**	**F**
Baseline	3	–	–	81.25	85.22	SVM	48	91.68	92.73	RF	48	–	–	
PCA	24	17	70.83	74.80	74.17	SVM	8	91.75	92.54	KNN	24	5.00	ETC	16
RFE	24	10	41.67	74.10	78.55	SVM	8	92.53	93.32	DT	16	1.75	AdaBoost	16
Autoencoder	24	16	66.67	74.89	74.38	SVM	8	89.86	90.57	GBT	24	5.00	MLP	16

Table 1 summarizes accuracy and F1-score ranges across dimensionality-reduction methods. T denotes total trials—each trial is one classifier on a reduced feature set. S counts trials achieving at least the baseline minimum, and SR (%) is the success rate. The MinScore block reports the lowest accuracy (ACC %) and its corresponding F1-score ($F1$ %), along with the classifier (C) and feature count (F). The MaxScore block gives the highest ACC % and $F1$ %, plus their C and F. Finally, MaxDiff records the largest drop in accuracy relative to the full-feature baseline, together with the classifier and feature count where it occurs.

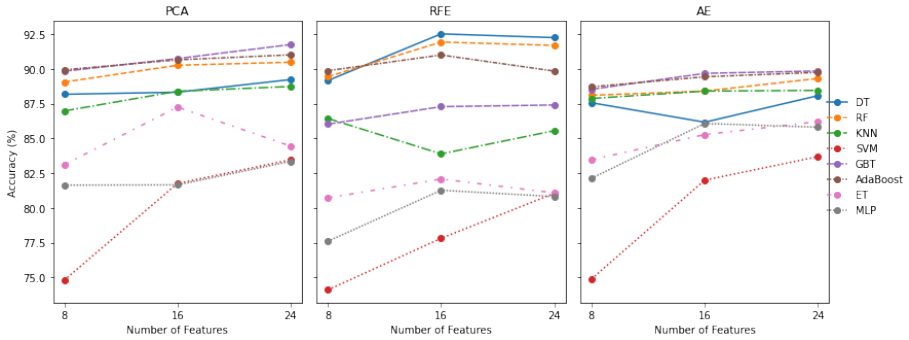

Fig. 1. Classifier accuracy vs. feature count for PCA, RFE, and Autoencoder.

Figure 1 shows accuracy versus retained feature count for all classifiers under PCA, RFE, and the deep autoencoder. As retention increases, accuracies improve across methods. RFE achieves the highest peak (92.5% at 16 features) and the smallest variation (≈ 1.8%), with Random Forest and k-NN achieving near 90%. PCA shows a sharper low at eight components (≈ 74.8% for SVM) but recovers near-baseline performance (> 91% for ensembles) once roughly half the variance is retained. The autoencoder follows similar trends, peaking around 89.9% (GBT) with a wider spread (≈ 5%) due to its nonlinear embeddings. Ensemble classifiers consistently outperform single models, and even eight-feature subsets preserve ≈ 75% accuracy, highlighting RFE's stability (in part due to its supervised nature), PCA's quick recovery, and the autoencoder's nonlinear benefits.

4. Conclusions

In this study, we assessed three dimensionality-reduction techniques—Principal Component Analysis, Recursive Feature Elimination, and a deep autoencoder—across eight classifiers for network intrusion detection.By evaluating each method at three reduction levels (8, 16, and 24 features) against a full-feature baseline, we identified clear trade-offs between accuracy, stabil-

ity, and computational cost.

Future work will explore hybrid schemes that combine RFE's robustness with PCA's efficiency, validate our findings on additional benchmarks such as CICIDS2017 and CSE-CIC-IDS2018, and investigate attention-augmented or lightweight convolutional autoencoders. We will also examine automated hyperparameter tuning and online reduction methods to support real-time intrusion detection pipelines under strict resource constraints.

References

[1] Di Mauro, M., Galatro, G., Fortino, G., Liotta, A.: Supervised feature selection techniques in network intrusion detection: A critical review. Engineering Applications of Artificial Intelligence 101, pp. 104216 (2021)

[2] Disha, R.A., Waheed, S.: Performance analysis of machine learning models for intrusion detection system using Gini Impurity-based Weighted Random Forest (GIWRF) feature selection technique. Cybersecurity 5(1) (2022)

[3] Li, P., Pei, Y., Li, J.: A comprehensive survey on design and application of autoencoder in deep learning. Applied Soft Computing 138, pp. 110176 (2023)

[4] Moustafa, N., Creech, G., Slay, J.: Big data analytics for intrusion detection system: statistical decision-making using finite dirichlet mixture models. In: Data Analytics and Decision Support for Cybersecurity, pp. 127–156. Springer, Cham (2017)

[5] Moustafa, N., Slay, J.: UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In: 2015 Military Communications and Information Systems Conference (MilCIS). pp. 1–6 (11 2015)

[6] Moustafa, N., Slay, J.: The evaluation of Network Anomaly Detection Systems: Statistical analysis of the UNSW-NB15 data set and the comparison with the KDD99 data set. Information Security Journal: A Global Perspective 25(1-3), pp. 18–31 (2016)

[7] Moustafa, N., Slay, J., Creech, G.: Novel Geometric Area Analysis Technique for Anomaly Detection Using Trapezoidal Area Estimation on Large-Scale Networks. IEEE Transactions on Big Data 5, pp. 481–494 (2019)

[8] Patil, R., Biradar, R., Ravi, V., Biradar, P., Ghosh, U.: Network traffic anomaly detection using PCA and BiGAN. Internet Technology Letters 5(1), pp. e235 (2022)

[9] Sarhan, M., Layeghy, S., Moustafa, N., Portmann, M.: NetFlow Datasets for Machine Learning-Based Network Intrusion Detection Systems. In: 10th EAI International Conference on Big Data Technologies and Applications. vol. 371, pp. 117–135. Springer (2021)

[10] Sharma, N.V., Yadav, N.S.: An optimal intrusion detection system using recursive feature elimination and ensemble of classifiers. Microprocessors and Microsystems 85, pp. 104293 (2021)

[11] Song, Y., Hyun, S., Cheong, Y.G.: Analysis of Autoencoders for Network Intrusion Detection. Sensors 21(13), pp. 4294 (2021)

[12] Zielosko, B., Stańczyk, U., Jabloński, K.: Construction of Features Ranking — Global Approach. In: B. Marcinkowski, A. Przybylek, A. Jarzebowicz, N. Iivari, E. Insfran, M. Lang, H. Linger, and C. Schneider (Eds.) Harnessing Opportunities: Reshaping ISD in the post-COVID-19 and Generative AI Era (ISD2024 Proceedings) (2024)