

Article

Cross-Domain Adversarial Alignment for Network Anomaly Detection Through Behavioral Embedding Enrichment

Cristian Salvador-Najar *  and Luis Julián Domínguez Pérez * 

Instituto Tecnológico y de Estudios Superiores de Occidente (ITESO), Anillo Periférico Sur Manuel Gómez Morín 8585, Santa María Tequexpan, Tlaquepaque 45604, Jalisco, Mexico

* Correspondence: cristian.najar@iteso.mx (C.S.-N.); luisjdominguezp@iteso.mx (L.J.D.P.)

Abstract

Detecting anomalies in network traffic is a central task in cybersecurity and digital infrastructure management. Traditional approaches rely on statistical models, rule-based systems, or machine learning techniques to identify deviations from expected patterns, but often face limitations in generalization across domains. This study proposes a cross-domain data enrichment framework that integrates behavioral embeddings with network traffic features through adversarial autoencoders. Each network traffic record is paired with the most similar behavioral profile embedding from user web activity data (Charles dataset) using cosine similarity, thereby providing contextual enrichment for anomaly detection. The proposed system comprises (i) behavioral profile clustering via autoencoder embeddings and (ii) cross-domain latent alignment through adversarial autoencoders, with a discriminator to enable feature fusion. A Deep Feedforward Neural Network trained on the enriched feature space achieves 97.17% accuracy, 96.95% precision, 97.34% recall, and 97.14% F1-score, with stable cross-validation performance (99.79% average accuracy across folds). Behavioral clustering quality is supported by a silhouette score of 0.86 and a Davies–Bouldin index of 0.57. To assess robustness and transferability, the framework was evaluated on the UNSW-NB15 and the CIC-IDS2017 datasets, where results confirmed consistent performance and reliability when compared to traffic-only baselines. This supports the feasibility of cross-domain alignment and shows that adversarial training enables stable feature integration without evidence of overfitting or memorization.

Keywords: anomaly detection; user profiling; representation learning; network traffic analysis; clustering-based security



Academic Editor: Leandros Maglaras

Received: 16 August 2025

Revised: 5 October 2025

Accepted: 10 October 2025

Published: 22 October 2025

Citation: Salvador-Najar, C.; Domínguez Pérez, L.J. Cross-Domain Adversarial Alignment for Network Anomaly Detection Through Behavioral Embedding Enrichment. *Computers* **2025**, *14*, 450. <https://doi.org/10.3390/computers14110450>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The exponential growth of the Internet in recent years, both in the number of users and the volume of data exchanged, has created an increased need to implement more robust security measures to protect networks against potential threats [1–4]. Also, the need for monitoring and management of network traffic increases as the volume of transmitted data grows [5]. In this context, anomaly detection emerges as a key strategy to identify unusual patterns in traffic that may signal cyberattacks or service interruptions [6]. The implementation of traffic monitoring systems is required to manage data flows and detect anomalies. These systems must perform continuous analysis of network activity and enable timely responses. Their design must align with the characteristics of the infrastructure, including data volume, network architecture, and security protocols. Effective traffic

monitoring depends on integration with scalable infrastructures capable of handling high loads without performance loss [7,8].

From the hardware perspective, Physically Unclonable Functions (PUFs) have been proposed to enhance device-level security in IoT and embedded systems. PUFs exploit microscopic manufacturing variations to generate unique challenge–response pairs, enabling hardware authentication, secure key generation, and IP protection without permanent secret storage [9–12]. Despite their value, PUFs remain vulnerable to modeling attacks, side-channel analysis, and environmental variations [13]. As a complementary line, software-based anomaly detection offers greater scalability and adaptability to network dynamics. Machine learning and deep learning have been widely explored, including wavelet-based multi-scale residual classifiers using stacked autoencoders for time–frequency feature extraction [14], as well as DNN, CNN, RNN, LSTM, GRU, and CNN-LSTM hybrids for intrusion detection [15,16]. Comparative studies on NSL-KDD show that recurrent models such as GRU achieve strong performance in binary classification tasks [17].

Anomaly detection frameworks that integrate ensemble learning with explainable AI have also been explored. For instance, one study addresses data imbalance through a hybrid sampling method combining SMOTE and K-means clustering; feature selection is performed using denoising autoencoders, XGBoost serves as the detector, and SHAP is employed for interpretability [18]. Graph neural networks have likewise been applied to IoT networks, merging GraphSAGE and graph attention networks to capture host–flow interactions, which improves anomaly detection while reducing false positives and negatives [19]. Another work employs a one-dimensional convolutional neural network that classifies traffic by TCP, UDP, and other protocols; chi-square guides feature selection, SMOTE handles class imbalance, and the UNSW-NB15 dataset is used for evaluation [8]. Regarding traditional machine learning, classifiers such as Decision Trees and Random Forests have been applied to detect DoS attacks in IoT networks, with feature selection via Genetic Algorithms and Correlation-based Feature Selection, and validation using the IoTID20 dataset [20].

Anomaly detection in IoT-enabled networks and smart environments has been widely studied due to the rapid growth of connected devices and the expanding attack surface. Early efforts centered on traditional IDS based on statistical or signature techniques. While rule-based systems efficiently detected known attack patterns, they struggled against zero-day or adaptive threats [21]. To address these limitations, machine learning was later introduced, with smart home models emphasizing feature selection and adaptability to heterogeneous IoT traffic [22]. These advances laid the groundwork for methods capable of capturing complex traffic behaviors. In parallel, scalable approaches for smart environments demonstrated that lightweight, device-specific classifiers can achieve up to 96% accuracy using only 32 flow-level features (e.g., byte counts, standard deviations), showing that flow-based fingerprinting can effectively replace deep packet inspection [23]. While traffic-level features (packet sizes, inter-arrival times, byte counts) have proven effective for anomaly detection in homogeneous environments [23], this work investigates whether cross-domain behavioral enrichment can prevent dataset-specific overfitting in heterogeneous network environments. Our contribution is not to replace efficient traffic-based methods, but to demonstrate the technical feasibility of adversarial alignment for feature space augmentation.

Recent research has increasingly focused on balancing accuracy with efficiency. Tree-based and ensemble methods have proven suitable for resource-constrained devices, with an IDS embedded in smart thermostats achieving real-time detection and low computational overhead [24]. Transformer-based architectures have also been applied to capture long-range dependencies, enabling smart home IDS to model temporal correlations across

device interactions with superior accuracy [25]. At the same time, federated learning and trustworthy AI have been adopted in 6G-connected smart buildings, where distributed training ensures privacy while preserving performance [26]. Transport protocol studies further highlight open challenges: a comparative review of QUIC and TCP uncovered protocol-specific vulnerabilities that IDS models must address [27], while anomaly detection tailored to QUIC traffic confirmed that protocol-aware features enhance detection accuracy [28].

It is important to note that, separately, user behavior profiling has also been explored as an alternative approach to identify anomalous activities in network environments. User behavior profiling, on its part, has become a critical tool for understanding network usage and strengthening its security as well [29–33]. The construction of user profiles enables the detection anomalous activities that could indicate malicious behavior [34,35]. This is especially relevant given the diversification of Internet usage, with an increase in activities such as video streaming, social media, and messaging apps that continuously modify traffic patterns [36]. Recent studies highlight that identifying patterns in user behavior improves the prediction of potential threats and optimizes network management by differentiating legitimate users from possible attackers [37,38]. A comparative summary of recent approaches, from traditional IDS models to hybrid and federated frameworks, is presented in Table 1.

While user behavior profiling and network anomaly detection have been extensively studied as separate domains, the specific integration of cross-domain behavioral enrichment with representation learning for network traffic data enhancement remains an emerging research area. This work addresses this gap by proposing a cross-domain data enrichment approach that integrates behavioral embeddings derived from user web activity with network traffic analysis. Rather than replacing existing traffic-based methods, our approach enriches network anomaly detection datasets through adversarial alignment between traffic data (UNSW-NB15 and CIC-IDS2017) [39,40] and user behavioral profiles captured via proxy monitoring. This enrichment prevents perfect memorization of dataset-specific patterns observed in traffic-only baselines. The main contribution of this work lies in the development of an adversarial cross-domain alignment methodology, showing that adversarial alignment produces feature representations with internal coherence to influence training dynamics compared to models relying solely on traffic features. By integrating these enriched representations, the deep learning model is trained to account for both traffic characteristics and user behavior patterns.

The rest of the paper is organized as follows. Section 2 presents essential background on user behavior profiling, network traffic, and anomalies needed to understand the proposed approach. Section 3 details the methodology, including data collection, user profiling, dataset enrichment, and model training. Section 4 reports the experimental results and analysis. Finally, Section 5 presents conclusions and directions for future research.

Table 1. Comparative analysis of network anomaly detection approaches.

Method	Year	Approach Type	Architecture	Dataset
Traditional ML [7]	2019–2021	Supervised SVM, RF, DT	Statistical features Rule-based detection	NSL-KDD KDD Cup 99
Multi-scale CNN [14]	2023	Deep Learning Hybrid	Wavelet transform + Stacked autoencoders	Custom dataset
Federated Learning [41]	2023	Distributed Collaborative	Privacy-preserving Cross-domain learning	Distributed datasets

Table 1. Cont.

Method	Year	Approach Type	Architecture	Dataset
Ensemble + XAI [18]	2024	Ensemble + Explainable AI	SMOTE + K-means + XGBoost + SHAP	UNSW-NB15 NSL-KDD
Graph Neural Networks [19]	2024	GNN Hybrid	GraphSAGE + Graph attention	IoT-specific datasets
GAN-based [38]	2024	Generative Adversarial	Synthetic data generation Sliding window adaptation	UNSW-NB15 Custom data
Dynamic Behavioral [42]	2024	User Profiling + ML	Real-time adaptation SDN integration	IoT networks
Proposed Method (This work)	2025	Hybrid User Profiling + DL	Cross-Domain profiling + Traffic enrichment	UNSW-NB15 + Charles Proxy

2. Background on Network Behavior and Anomalies

2.1. User Behavior Modeling for Profiling

User behavior modeling refers to the process of representing and analyzing patterns of actions performed by users within a system or network. In cybersecurity, this modeling facilitates the identification of deviations that may signal malicious activity or system misuse [43,44]. The underlying assumption is that normal user behavior follows certain patterns that can be captured and represented mathematically, while anomalies indicate potential threats.

In this study, user behavior is represented as a fixed-length numerical vector. Each dimension of the vector corresponds to the count of HTTP or HTTPS connections initiated to a monitored domain or application. Formally, for each user $u \in \mathcal{U}$, the behavior vector is defined as

$$\mathbf{v}_u = [v_1, v_2, \dots, v_n] \in \mathbb{R}^n, \quad (1)$$

where v_i denotes the number of requests made to service i and n is the total number of monitored services. Each service i corresponds to a specific platform (e.g., Teams, Thingiverse, Multisim, etc.) identified in the collected traffic dataset.

The vector $\mathbf{v}_u$ is constructed from raw traffic logs obtained using Charles Proxy. These logs include timestamps, source IP addresses, and domain names or endpoints accessed by each user. Each log entry is assigned to a domain category, and requests are aggregated into a single vector per user. This vector captures the activity of the user distribution across services during the observation window.

To model user behavior using clustering, the set of user behavior vectors $\{\mathbf{v}_u \mid u \in \mathcal{U}\}$ is processed by a clustering algorithm that assigns each vector to one of the clusters $\{C_1, C_2, \dots, C_k\}$. Each cluster C_j is represented by its centroid $\mu_j \in \mathbb{R}^n$, computed as the mean of the vectors within the cluster:

$$\mu_j = \frac{1}{|C_j|} \sum_{\mathbf{v}_u \in C_j} \mathbf{v}_u. \quad (2)$$

The assignment of a user behavior vector $\mathbf{v}_u$ to a cluster is determined by finding the centroid closest to it according to a distance metric $d(\cdot, \cdot)$, typically the Euclidean distance:

$$\hat{C}(\mathbf{v}_u) = \arg \min_{j=1, \dots, k} d(\mathbf{v}_u, \mu_j). \quad (3)$$

Anomaly detection is performed by comparing the distance between the user behavior vector $\mathbf{v}_u$ and its assigned cluster centroid $\mu_{\hat{C}(\mathbf{v}_u)}$ to a predefined threshold τ . The cluster

assignment function $\hat{C}(\mathbf{v}_u)$ maps each user vector to its nearest cluster based on Euclidean distance in the latent embedding space. Specifically, the distance calculation is performed as

$$d(\mathbf{v}_u, \mu_{\hat{C}(\mathbf{v}_u)}) = \|\mathbf{v}_u - \mu_{\hat{C}(\mathbf{v}_u)}\|_2 \quad (4)$$

where $\|\cdot\|_2$ denotes the Euclidean norm. The threshold parameter τ is determined through empirical validation using cross-validation on the training dataset. In practice, τ is set as the 95th percentile of distances between normal user vectors and their respective cluster centroids, ensuring that approximately 5% of normal behaviors are classified as potential anomalies, which provides a balance between detection sensitivity and false positive rates. This threshold selection approach is consistent with standard anomaly detection practices and can be adjusted based on specific security requirements.

The anomaly detection rule is then formulated as

$$d(\mathbf{v}_u, \mu_{\hat{C}(\mathbf{v}_u)}) > \tau \implies \text{anomaly} \quad (5)$$

The function notation $[\mathbf{v}_u; \mu_{\hat{C}(\mathbf{v}_u)}]$ represents the concatenation of the user behavior vector with its corresponding cluster centroid, creating an enriched feature vector that combines individual behavioral patterns with group-level behavioral characteristics. This concatenation operation enables the integration of both instance-specific and cluster-representative information, providing a more comprehensive behavioral context for anomaly detection.

The clustering algorithm takes as input the user behavior vectors $\{\mathbf{v}_u\}_{u=1}^N$ extracted from the autoencoder latent space and outputs cluster assignments $\hat{C}(\mathbf{v}_u)$ along with anomaly decisions based on the distance thresholds. The parameters of the clustering process, including the number of clusters and linkage criteria, are determined through clustering validation metrics such as the silhouette score and Davies–Bouldin index to ensure optimal cluster quality and separation.

This representation framework allows the seamless integration of behavioral features with traditional traffic-based features, enhancing the semantic scope of anomaly detection by providing contextual information about user patterns. Understanding how behavioral features align with packet-level characteristics requires a comprehensive review of network traffic composition and its potential for anomalous behavior manifestation. The following subsection introduces the concept of network traffic and its fundamental role in the emergence and identification of network-based anomalies.

2.2. Network Traffic and Anomalies

Network traffic is the flow of data packets exchanged between devices in a communication network. This traffic includes data governed by protocols such as the Transmission Control Protocol (TCP), User Datagram Protocol (UDP), and others [45]. TCP manages reliable, ordered, and error-checked delivery of data. UDP provides connectionless, low-latency communication with minimal overhead. The variety of protocols and volume of data require analysis to maintain network performance, reliability, and security. Anomalies in network traffic are deviations from baseline patterns. These deviations may result from faults, misconfigurations, or malicious activities. Detecting anomalies supports identification of cyberattacks, unauthorized access, or service interruptions that affect network resources [46].

Generally, anomalies occur in different forms depending on attack vectors and objectives. Denial of Service (DoS) attacks overload targets with excessive traffic or requests to disrupt service availability. Reconnaissance attacks scan networks to gather information about system configurations and vulnerabilities before further attacks. Exploits use software or protocol vulnerabilities to gain unauthorized access or escalate privileges. Mal-

ware propagation via worms, backdoors, and shellcode enables attackers to compromise systems, maintain access, and execute code remotely. Fuzzing attacks send malformed or unexpected inputs to expose security weaknesses. Generic attacks include various attack methods that do not fit other categories [46,47].

Network traffic can be described by a set of features extracted from packets, flows, or sessions. These features include packet size, inter-arrival time, protocol type, port numbers, and statistical aggregates such as mean duration or byte count. Let $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_m\}$ denote a dataset of m network traffic records, where each $\mathbf{x}_i \in \mathbb{R}^n$ is a feature vector representing traffic characteristics. Anomaly detection in network traffic consists of identifying patterns that deviate from a defined notion of normality. This process supports intrusion detection systems and contributes to the protection of networked environments. The formal goal of anomaly detection is to build a function

$$f: \mathbb{R}^n \rightarrow \{0, 1\} \quad (6)$$

that assigns a binary label to an input feature vector $\mathbf{x} \in \mathbb{R}^n$, where $f(\mathbf{x}) = 1$ indicates an anomaly and $f(\mathbf{x}) = 0$ indicates normal behavior. Such a function can be learned using expert rules, ensemble methods, or deep learning. In this work, we use a deep learning model that combines user behavior and traffic patterns to enhance anomaly detection accuracy and adaptability.

To incorporate user behavior, let $\mathcal{U} = \{\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_m\}$ be a set of user profile feature vectors associated with each network traffic record $\mathbf{x}_i$, where $\mathbf{u}_i \in \mathbb{R}^p$ represents behavioral features. In this work, we use a deep learning model that combines traffic features $\mathbf{x}_i$ and user behavior features $\mathbf{u}_i$ to enhance anomaly detection accuracy and adaptability. Behavior profiling combined with network anomaly detection datasets is used to classify anomalous network activity, as described in the following section.

The cross-domain aspect of profiling is achieved at the record level: each UNSW-NB15 traffic sample is paired with the most similar profile embedding from the Charles dataset using cosine similarity. This instance-specific assignment provides contextual behavioral information for every network record, enhancing the ability of the anomaly detection model to distinguish normal from anomalous traffic.

3. Methodology

3.1. Data Extraction

This study uses three data sources: self-obtained network traffic for user profiling, the UNSW-NB15 dataset for attack pattern detection, and the CIC-IDS-2017 dataset. In this study, the binary versions of both datasets were used, as in [48]. This setup aligns with the behavioral enrichment process using the Charles dataset, whose embeddings capture general traffic patterns rather than attack-specific features. By keeping all attack types in a single class, each traffic record receives meaningful enrichment without introducing noise or inconsistencies. This binary setting preserves the diversity of attack instances while ensuring that the enrichment is consistent across all samples. Charles Proxy (v5.0.3, XK72 Ltd., Wellington, New Zealand; available online: <https://www.charlesproxy.com>, accessed on 20 September 2025) helps capture and examine HTTP and HTTPS traffic between a client (like a browser or app) and a server, making it useful for observing network activity in real-time. A particular set of sites was monitored, and its log is shown in Table 2.

The first dataset was self-collected specifically for this study by monitoring network traffic from 90 students, each observed for 30 min. Charles Proxy was selected over alternatives (tcpdump, Wireshark) for its ability to seamlessly capture application-layer traffic without disrupting user sessions or requiring network infrastructure modifications. This approach captures user interaction patterns with specific web services (Table 2) that reflect

genuine behavioral diversity across different user roles. The total observation time was limited by the software license, which allowed only 30 min of use. The students were divided into three groups based on their academic and professional backgrounds: 20 network analysts, 10 programmers, and 60 students from a mechatronics program. This division allowed the study to explore different user behaviors and their network traffic patterns. Charles Proxy was used to capture network traffic from each user. The traffic included information about visited web pages, application usage, and established connections. This data was used to build user profiles based on traffic type. The captured data included HTTP headers, cookies, URLs, and response times. The use of a proxy enabled traffic capture without modifying the network or interrupting user sessions. Network activity from different user groups was collected to support comparison across usage patterns. The collected data was used for user profiling.

All 90 users provided consent to participate in the study. Each session lasted approximately 30 min, resulting in approximately 2700 min (45 h) of total observation time, covering diverse activities such as web browsing, application usage, and network connections. This dataset was then used to build user profiles based on traffic type, allowing comparison across different user groups and supporting the anomaly detection experiments.

For the UNSW-NB15 dataset, the full dataset was used with a 70/30 training/testing split, removing identifying features (User, IP address, category), while retaining the remaining features were used for modeling. These details ensure ethical compliance and sufficient information for reproducibility of the anomaly detection experiments. In parallel, the CIC-IDS-2017 dataset was also prepared to provide complementary validation. Following a stratified subsampling strategy, a representative 5% subset was extracted while preserving the original benign/anomalous distribution. Only numerical attributes were retained, missing values and infinities were imputed using median values, and all features were standardized before being converted into PyTorch (v2.8.0) tensors for model training. This preparation ensured that CIC-IDS-2017 could be consistently integrated with the UNSW-NB15 experimental framework while maintaining computational feasibility.

UNSW-NB15 was selected over alternative datasets (NSL-KDD, KDD Cup 99, IoTID20) for several methodological reasons. Firstly, the binary version of UNSW-NB15 was used as in [48], which provides a balanced normal/anomalous classification structure, essential for evaluating regularization effects without the complexity of multi-class attack categorization. This setup aligns with the behavioral enrichment process using the Charles dataset, whose embeddings capture general traffic patterns rather than attack-specific features. By keeping all attack types in a single class, each traffic record receives meaningful enrichment without introducing noise or inconsistencies that could arise from attempting to match behavioral profiles to specific attack categories. Secondly, UNSW-NB15 offers modern network traffic characteristics that are more representative of current network environments compared to legacy datasets like KDD Cup 99. The dataset includes contemporary attack vectors and normal traffic patterns that facilitate realistic evaluation of cross-domain alignment methodologies. Thirdly, the dataset's feature engineering includes both basic network statistics and advanced flow-based features, providing sufficient complexity to demonstrate the regularization benefits of behavioral enrichment while maintaining comparability with the existing literature.

In addition, the CIC-IDS-2017 dataset was incorporated to strengthen external validation and assess the robustness of the proposed methodology under different traffic distributions. Unlike UNSW-NB15, CIC-IDS-2017 contains a larger set of heterogeneous flows and diverse attack scenarios, offering a complementary perspective on anomaly detection performance. A stratified subsample was prepared to ensure computational feasibility while preserving the original benign/anomalous balance. By combining UNSW-NB15 as

the primary benchmark with CIC-IDS-2017 as a secondary validation set, the experimental design not only leverages the methodological clarity of UNSW-NB15 but also tests the generalization capacity of the adversarial alignment approach across datasets with distinct traffic and attack profiles.

Table 2. List of services accessed by participants during data collection using Charles Proxy. This dataset captures HTTP/HTTPS traffic generated by 90 users across various platforms, reflecting real-world user behavior for profiling and network analysis. Each entry corresponds to a monitored service or platform visited during the session.

Site	Description
thingiverse	3D design and model sharing platform
multisim	Electronic circuit simulation tool
canvas	Online learning management platform
wikipedia	Free online encyclopedia with collaborative content
netacad	Cisco’s educational platform for learning networking
arduino	Development environment for electronic projects
matlab	Software for mathematical analysis and programming
stackoverflow	Community of questions and answers for programmers
SevOne	Network and IT performance monitoring solution
PacketTracer	Cisco network simulator for learning networking concepts
Spectrum	Network and Performance Management Platform
ServiceNow	IT service automation platform
Visual Studio	Integrated development environment from Microsoft
Udemy	Online learning platform with courses in various fields
Xcode	Development environment for iOS and macOS applications
Github	Version control and development platform based on Git

The integration of the Charles Proxy data with other datasets, the UNSW-NB15 and the CIC-IDS-2017 datasets, is described in the following subsections. The goal of this integration is not to match the specific content of visited sites but to align behavioral patterns. Autoencoders and a discriminator are used to produce latent representations of each dataset and align them through adversarial training. This process maps both sources into a shared latent space, allowing the model to learn from both traffic and behavior without requiring semantic correspondence. The enrichment does not modify labels or alter the meaning of the datasets but introduces additional context based on user activity patterns. We first describe the computed user profiling, followed by the merging of the two datasets.

3.2. Data Preparation

After data extraction, preprocessing steps were performed to prepare the datasets for anomaly detection. Feature extraction was performed to convert traffic and behavioral logs into numerical representations. User profiles were created by clustering embeddings generated by autoencoders trained with adversarial learning to align the latent gaps between the datasets. The UNSW-NB15 dataset was enriched by incorporating these behavioral embeddings, adding user context to the network traffic data. Finally, the combined dataset was normalized and split into training and test subsets to facilitate model training and evaluation. These steps prepared the data for building and validating the detection system.

3.2.1. Behavioral Data Preparation

Behavioral data preparation involved calculating the frequency of use for each service and the activity entropy for each user, followed by dimensionality reduction using Principal

Component Analysis (PCA) [49]. First, the frequency of use for each service was calculated by dividing each service interaction by the total number of interactions for each user [50]. Normalizing the frequencies standardized the data, allowing the comparison of users with different activity levels. This process enabled the analysis of user behavior independently of the total number of interactions, allowing comparison across users.

The formula for calculating the normalized frequency of each service is as follows:

$$f_{ij} = \frac{x_{ij}}{\sum_{j=1}^n x_{ij}}, \quad (7)$$

where f_{ij} is the normalized frequency of service j for user i , x_{ij} is the raw count of interactions for service j by user i , and $\sum_{j=1}^n x_{ij}$ is the total number of interactions for user i . Next, the activity entropy for each user was calculated to measure the randomness in their interactions.

Activity entropy is a metric that reflects how unpredictable or varied actions from an user are [51]. A higher entropy value indicates that a behavior of a particular user is more random, while a lower entropy suggests more consistent behavior. This metric was important for profiling users, as it allowed us to distinguish between those with varied behavior and those with more focused patterns. By applying the formula of the Shannon entropy, we were able to capture the degree of randomness in users activity, providing a deeper insight into their interaction patterns. The formula for calculating entropy for each user is

$$H_i = - \sum_{j=1}^n p_{ij} \cdot \log(p_{ij}), \quad (8)$$

where H_i is the entropy for user i , p_{ij} is the probability of interaction with service j by user i , and n is the number of services. The probability p_{ij} is calculated as

$$p_{ij} = \frac{x_{ij}}{\sum_{j=1}^n x_{ij}}. \quad (9)$$

Finally, PCA was applied to reduce dimensionality by extracting the two principal components. These components were added to the existing dataset rather than replacing the original features. This approach retained all original feature details while incorporating the PCA components, which summarize the key variance in the data. Adding PCA components provided complementary information that can improve the modeling of user behavior patterns and anomaly detection, enabling the model to leverage both detailed and aggregated feature representations.

The final behavioral dataset includes several features per user. These are the normalized frequency of interaction with each monitored service, representing the proportion of accesses to each resource relative to the total activity. Each user profile also contains a value for activity entropy, calculated using Shannon's entropy formula to measure variability in behavior. Two principal components obtained from PCA were added to reduce dimensionality and summarize patterns of variation in user activity. The dataset combines interaction frequencies, entropy values, and PCA components, providing input for modeling and anomaly detection.

Behavior Profiling

The behavior profiling phase used clustering techniques to group data based on patterns in user behavior. After normalization, different clustering algorithms were applied and compared to identify distinct user behavior groups and select the best approach for profiling. Various unsupervised learning techniques were applied to group the data based

on patterns within the user behavior. The data, after being normalized, was subjected to different clustering algorithms to identify distinct patterns of user behavior. The first approach involved K-means [52] and agglomerative clustering [53] with autoencoders, and DBSCAN [54]. K-means and agglomerative clustering require the specification of the number of clusters, hence various values for the number of clusters were tested to identify the optimal count. For K-means and agglomerative clustering, a range of clusters from 2 to 10 was explored, and the optimal clustering configurations were determined using the silhouette score [55], which measures the quality of the clusters. DBSCAN, in contrast, does not require a predefined number of clusters but instead uses the parameter *eps* to define the maximum distance between points within a cluster. A series of values for *eps* were tested, and the best configuration was chosen based on the silhouette score as well. In addition to the silhouette score, we calculated an additional metric, the Davies–Bouldin index, to evaluate cluster quality. This ensures that the clusters reflect actual behavioral patterns rather than arbitrary divisions. With this information, the clustering steps are fully connected to the parameters, the dimensionality reduction, and the similarity measurement in the real data.

Clustering parameters and the calculation of distances between vectors are defined to ensure that they accurately capture the underlying structure of the data in real-world networks. As stated, after dimensionality reduction with an autoencoder, each user is represented as a latent vector v_u . The centroid of each cluster u_j is calculated as the mean of the vectors of the users assigned to that cluster. The similarity between a user and its cluster is measured using the Euclidean distance in the latent space as in Equation (4).

The behavioral profiling methodology used in this study follows a multi-stage approach to extract patterns from user interaction data. First, a preprocessing phase is performed in which the data is normalized to ensure analytical consistency. Normalization prevents features with larger numerical ranges from dominating the analysis, such as session duration versus click frequency.

After normalization, a hybrid model is applied that combines dimensionality reduction using autoencoders and unsupervised clustering with agglomerative algorithms. This combination addresses the complexity of high-dimensional behavioral data and the nonlinear relationships between features.

The autoencoder compresses the input data into a lower-dimensional latent space, minimizing reconstruction error and preserving essential patterns. The resulting latent representation is processed by an agglomerative clustering algorithm, which hierarchically clusters the data based on similarity in the reduced space. This approach identifies hierarchical structures in behavioral patterns and assigns group labels based on actual similarity.

The integration of the autoencoder with clustering allows for the identification of behavioral profiles that would not be detectable using linear methods or direct clustering on the original data. The resulting profiles represent user archetypes with distinct interaction and behavior patterns, useful for network security analysis.

The entire process is regulated by hyperparameters that control the autoencoder architecture, training dynamics, and clustering criteria. These parameters were optimized through experimentation and are detailed in Table 3. Figure 1 shows the data flow from input to the fusion of behavioral profiles with the UNSW-NB15 dataset.

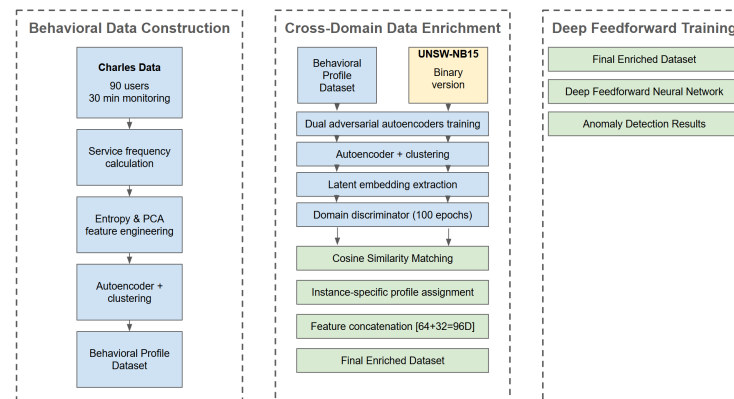


Figure 1. Cross-domain enrichment framework for network anomaly detection. The pipeline illustrates three main stages: (i) construction of behavioral profiles from monitored web activity, (ii) cross-domain data enrichment where embeddings are aligned with UNSW-NB15 traffic records through adversarial autoencoders and cosine similarity matching, and (iii) training of a deep feedforward neural network on the enriched dataset. This process demonstrates how behavioral embeddings complement traffic features to improve generalization in anomaly detection.

The architecture of the autoencoder consisted of an input layer, a hidden layer that produced the reduced representation (the latent space), and a final output layer that reconstructed the original input data. In this case, the autoencoder was configured to reduce the dimensionality of the data to two features. The model was trained to minimize the reconstruction error using a mean squared error loss function. Once the autoencoder was trained, the encoded data (the compressed representation) was extracted and used as the input for clustering. This enabled us to reduce data complexity while preserving its essential patterns.

The next step involved applying clustering algorithms to the encoded data, testing various numbers of clusters to find the optimal clustering solution. The objective was to determine both the best number of clusters and the algorithm that produced the most accurate grouping. For each clustering configuration, several evaluation metrics were calculated to assess clustering quality. These included the silhouette score, which measures how well each data point fits into its assigned cluster; the Calinski–Harabasz index, which evaluates cluster cohesion and separation; and the Davies–Bouldin index, which assesses cluster similarity. Together, these metrics helped identify the clustering solution with the best performance. Once the dataset of user profiles is computed, it is merged with the UNSW-NB15 dataset using adversarial training.

Table 3. Parameters used in the hybrid model (autoencoder and agglomerative clustering).

Parameter	Value
Data Preprocessing	Data normalization
Dimensionality Reduction (Autoencoder)	32 (latent dimension)
Input Layer (Autoencoder)	Input: (n_features,)
Hidden Layer (Autoencoder)	Dense (64, activation = 'relu')
Latent Layer (Autoencoder)	Dense (32, activation = 'linear')
Output Layer (Autoencoder)	Dense (n_features, activation = 'sigmoid')
Optimization (Autoencoder)	Adam
Loss Function (Autoencoder)	Mean Squared Error
Epochs (Autoencoder)	100
Batch Size (Autoencoder)	256
Number of Clusters (Agglomerative Clustering)	2
Linkage (Agglomerative Clustering)	Ward

3.3. Dataset Enrichment via Adversarial Training

The dataset enrichment process integrates behavioral data embeddings from the Charles Proxy dataset with the UNSW-NB15 attack dataset. This integration aims to enhance anomaly detection by providing richer context to each network traffic record. First, numerical features were extracted separately from both datasets and standardized to normalize their ranges. Two separate autoencoder models were defined: one for the UNSW-NB15 dataset and one for the Charles dataset. Each autoencoder compresses input features into a latent embedding of dimension 32 and reconstructs the input from this latent space. A discriminator model was also defined to differentiate between latent embeddings generated by each autoencoder, facilitating adversarial training to align the latent spaces across datasets. The same procedure was also applied to the CIC-IDS-2017 dataset, ensuring methodological consistency across both benchmarks.

The models were trained adversarially over 100 epochs. In each training iteration, the discriminator learned to classify latent embeddings as either originating from the UNSW-NB15 or Charles dataset. Simultaneously, the autoencoders optimized their reconstruction losses and tried to fool the discriminator by aligning their latent embeddings. The combined loss function for the autoencoders included a mean squared error term for reconstruction and a binary cross-entropy term weighted by 0.1 for adversarial loss. The key training parameters are summarized in Table 4. The encoder and decoder architectures used in the adversarial autoencoder training are the same as those reported in Table 3. Convergence during training was monitored by observing both the reconstruction loss and the adversarial loss over epochs, ensuring that the losses stabilized before completing the 100 training epochs. This guarantees that the autoencoders effectively learned to reconstruct inputs while aligning their latent embeddings across datasets.

After training, latent embeddings were extracted from both datasets. For each UNSW-NB15 embedding u_i , the most similar embedding c_j from the Charles dataset was identified using cosine similarity:

$$\text{sim}(u_i, c_j) = \frac{u_i \cdot c_j}{\|u_i\| \|c_j\|} \quad (10)$$

This similarity score measures the alignment between the network traffic patterns represented in u_i and the behavioral profiles represented in each c_j . The embedding c_j with the highest similarity was selected and concatenated with u_i to form the enriched representation:

$$\tilde{u}_i = [u_i; c_j] \quad (11)$$

This process produces a joint feature vector that integrates network traffic and behavioral information for each instance. The enrichment is instance-specific, ensuring that each UNSW-NB15 record is paired with the most contextually relevant behavioral profile. By selecting embeddings based on similarity, the approach preserves the variability of behavioral patterns, provides richer contextual information for each network record, and facilitates the detection of subtle deviations indicative of anomalous or malicious activity. The final enriched dataset was stored with the original labels for subsequent model training and evaluation.

Table 4. Training parameters used in adversarial autoencoder dataset enrichment.

Parameter	Value
Batch size	128
Latent dimension	32
Learning rate	0.001
Number of epochs	100
Reconstruction loss	Mean Squared Error
Adversarial loss weight	0.1
Optimizer	Adam

3.4. Deep Feedforward Neural Network for Anomaly Detection

The classification model for anomaly detection is a Deep Feedforward Neural Network (DFNN) trained on the enriched dataset. The architecture consists of an input layer that receives 96 features, corresponding to the concatenation of a 64-dimensional latent representation from the UNSW-NB15 dataset and a 32-dimensional behavioral profile from the Charles dataset. The input is passed through three fully connected hidden layers with 128, 64, and 32 units, respectively. Each hidden layer uses the ReLU activation function. The final output layer is a single neuron with a sigmoid activation function to perform binary classification of network traffic as either benign or malicious. The model was trained using binary cross-entropy as the loss function and the Adam optimizer. The training process was conducted over multiple epochs using mini-batches. Table 5 summarizes the training parameters used.

Table 5. Detailed architecture and training parameters for the Deep Feedforward Neural Network classifier.

Parameter	Value
Input dimension	96
Number of hidden layers	3
Hidden layer sizes	128, 64, 32
Activation functions	ReLU (hidden), Sigmoid (output)
Dropout rate	0.3 (after each hidden layer)
Loss function	Binary Cross-Entropy
Optimizer	Adam
Learning rate	1×10^{-3}
Batch size	128
Number of epochs	20

3.5. Evaluation Framework

The evaluation framework in this study is designed to assess three main aspects of our methodology: the quality of clustering for behavioral profiling, the effectiveness of dataset enrichment through domain alignment, and the performance of the classifier in detecting anomalies. This comprehensive evaluation directly supports the motivation of our work by demonstrating that integrating user behavioral profiles improves anomaly detection and achieves better results than traditional approaches.

For clustering evaluation, we employ multiple metrics to measure the quality of the clusters and to justify parameter selection. The silhouette score quantifies cohesion within clusters and separation between clusters, helping to identify the optimal number of clusters. The Calinski–Harabasz index evaluates the overall separation among clusters, while the Davies–Bouldin index measures the similarity between clusters, with lower values indicating better-defined groups. The Dunn index assesses the balance between intra-cluster compactness and inter-cluster separation. Together, these metrics ensure that

the clusters reflect genuine behavioral patterns rather than arbitrary divisions. By applying an autoencoder for dimensionality reduction, the latent representations capture essential features and reduce noise, which improves the separation and stability of clusters across different configurations.

The impact of dataset enrichment is evaluated by integrating behavioral profile embeddings with the UNSW-NB15 dataset. Visualizations using t-SNE are employed to verify the separation between domains and the compactness of the enriched embeddings. Additionally, latent structure analysis assesses whether the combined representations improve the feature space available to the classifier, facilitating more effective learning and anomaly detection.

Anomaly classification performance is evaluated using accuracy, precision, recall, and F1-score, which provide a comprehensive view of the model performance in detecting both normal and anomalous traffic. Cross validation is performed to ensure the stability and generalizability of the results. The confusion matrix provides detailed insights into true positives, false positives, true negatives, and false negatives, allowing a deeper understanding of the classifier performance. The use of these metrics aligns with the motivation of the study, confirming that behavioral profiling contributes to more accurate detection of anomalous activity [56,57].

To contextualize the results, we compare our hybrid approach (enriched dataset with clustering and DNN classifier) with the baseline model trained on the original dataset without enrichment. Additionally, standard approaches reported in the literature are included for comparison, such as K-means combined with a deep feedforward network, DBSCAN, and agglomerative clustering without autoencoder embeddings. This allows us to demonstrate that latent behavioral representations improve cluster quality and anomaly detection performance relative to both our baseline and typical existing methods.

Finally, all clustering and classifier parameters, including the number of clusters, *eps* for DBSCAN, learning rate, batch size, and number of epochs, are selected based on validation metrics. Cross-validation and multiple evaluation metrics are applied to ensure robust performance and to prevent overfitting, guaranteeing that the methodology generalizes effectively to unseen data.

4. Experimental Results

This section presents the experimental results in three stages. The first part shows the outcomes of the user profiling process based on clustering applied to normalized traffic data. The second part presents the results of merging the user profiling embeddings with the UNSW-NB15 and the CIC-IDS2017 dataset using adversarial training. The third part includes the classification results obtained by training a deep feedforward neural network on the combined dataset. Each stage is evaluated separately to examine its contribution to the overall anomaly detection performance.

4.1. Behavior Profiling

The ultimate goal of this stage is to construct a behavioral dataset from the raw interaction logs collected with Charles Proxy. Since the proxy outputs low-level records of web service usage, cookies, headers, and connection events, these data must be transformed into higher-level user profiles that summarize recurring behavioral patterns. Clustering provides a principled way to group users with similar interaction habits, enabling the generation of representative embeddings that can later enrich the UNSW-NB15 traffic records. To ensure that these clusters capture genuine behavioral regularities rather than artifacts of the algorithms, we rely on several complementary quality metrics. The silhouette score indicates whether individual users are well matched to their assigned cluster

compared to neighboring ones, reflecting cohesion and separation. The Calinski–Harabasz index measures the tradeoff between intra-cluster compactness and inter-cluster dispersion, with higher values supporting well-separated user archetypes. The Davies–Bouldin index evaluates the average similarity between clusters, where lower values imply greater distinction among behavioral groups. Finally, the Dunn index assesses the balance between the tightness of each cluster and the distance between clusters, where higher values reflect more robust separation. Taken together, these metrics guide the selection of the clustering configuration, ensuring that the resulting behavioral dataset encodes consistent user profiles that can be meaningfully integrated into anomaly detection experiments.

The clustering results from various methods without using the hybrid model (autoencoder and agglomerative clustering) provide valuable insights into the performance of traditional clustering algorithms. The results are shown in Figure 2. K-means clustering yields silhouette scores ranging from 0.5048 to 0.4159, indicating a moderate to weak clustering structure. A higher silhouette score generally suggests that the clusters are well-separated, while a lower score implies that the clusters are less distinct. The scores decrease as the number of clusters increases, but the performance does not degrade drastically, suggesting a relatively stable clustering behavior across different values of k .

Clustering results from traditional methods without using the hybrid model (Autoencoder and agglomerative clustering) were analyzed to evaluate the performance of K-means, agglomerative clustering, and DBSCAN. Figure 2 presents the silhouette scores for each method across a range of cluster numbers.

The silhouette scores for agglomerative clustering range from 0.4739 to 0.3913, exhibiting a trend comparable to K-means clustering. The scores are slightly lower on average than those for K-means, and the performance does not improve significantly with increasing cluster numbers. However, it does not partition the data more effectively than K-means when applied directly to the original data. In contrast, DBSCAN produces a range of scores from 0.4286 to -1, which includes negative values for some parameter choices. The negative silhouette scores indicate that DBSCAN has failed to cluster some data points correctly, likely due to its inability to identify meaningful clusters or because of a high number of noise points. This suggests that DBSCAN may not be as reliable in this case, possibly due to challenges in determining the appropriate density threshold for the data.

When using the hybrid model, the autoencoder reduces the dimensionality of the data to a 2D latent space, allowing the clustering algorithm to focus on the essential features of the data. This approach significantly improves the clustering performance. Agglomerative clustering with the latent features from the autoencoder achieved much higher and more consistent silhouette scores, ranging from 0.7861 to 0.8632. These improved scores show a significant enhancement over traditional clustering methods. The higher silhouette scores indicate that the clusters are better defined, with the hybrid model effectively capturing the underlying structure of the data.

The results demonstrate that the reduced latent representation generated by the Autoencoder enables more effective cluster separation. The clustering performance is more stable across different numbers of clusters, achieving the highest silhouette score with an optimal number of clusters. This suggests that dimensionality reduction helps uncover the inherent structure of the data, enabling more distinct and meaningful clusters.

The hybrid model using an autoencoder followed by agglomerative clustering shows higher performance than K-means, DBSCAN, and agglomerative clustering without dimensionality reduction. The autoencoder extracts a low-dimensional representation of the input, which improves the separation of groups when used as input for clustering. The model achieves higher silhouette scores across tested configurations. Additional clustering metrics support this result. The Calinski–Harabasz index reaches 195.30, indicating separa-

tion among groups. The Davies–Bouldin index reaches 0.5775, showing limited overlap between groups. The Dunn index reaches 0.0729, showing that intra-group compactness is not optimal. These metrics reflect consistent clustering performance and provide a basis for comparison across different models.

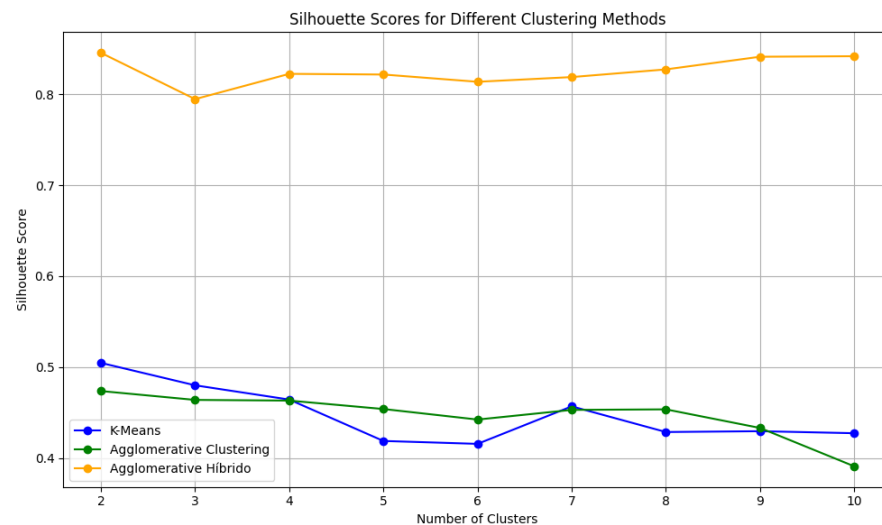


Figure 2. Comparison of silhouette scores for different clustering methods. This figure compares the silhouette scores for K-means, agglomerative clustering, and hybrid agglomerative clustering (autoencoder + agglomerative clustering) across a range of cluster numbers (2 to 10). The plot shows how each method performs in terms of cluster separation, with higher silhouette scores indicating better-defined clusters. The hybrid agglomerative clustering approach demonstrates superior performance, achieving consistently higher scores compared to the traditional K-means and agglomerative clustering methods.

Although the Davies–Bouldin index is relatively low, it is not minimal due to partial overlap between clusters corresponding to users with intermediate behavior patterns. For example, some students in the mechatronics group exhibit both programming and networking activities, creating latent embeddings that lie between the distinct behavioral groups, increasing the intra-cluster distance relative to inter-cluster separation.

The Calinski–Harabasz index of 195.30 indicates a strong between-cluster separation relative to within-cluster dispersion, confirming that the latent features extracted by the autoencoder highlight distinctive behavioral patterns. The Dunn index of 0.0729 suggests that although clusters are well-separated, intra-cluster compactness could be improved, reflecting variability among users within the same behavioral group.

Overall, the hybrid autoencoder–agglomerative clustering approach demonstrates superior clustering quality compared to traditional methods and PCA baselines. While the silhouette score indicates well-defined clusters, the moderate Davies–Bouldin and Dunn indices reflect the presence of users with mixed or transitional behaviors, which naturally reduces perfect cluster separation. These results confirm that dimensionality reduction via autoencoders captures the non-linear structures in user behavior, enabling more meaningful profiling for downstream anomaly detection.

The clustering analysis validates the feasibility of transforming raw Charles Proxy logs into a structured behavioral dataset suitable for cross-domain enrichment. Traditional methods such as K-means and DBSCAN showed limited separation capacity, highlighting the challenge of capturing complex user interaction patterns directly from high-dimensional data. By contrast, the hybrid autoencoder–agglomerative clustering approach achieved substantially higher and more consistent performance across all evaluation metrics, confirming that dimensionality reduction uncovers latent behavioral structures otherwise hidden in the

raw traffic. The combination of silhouette, Calinski–Harabasz, Davies–Bouldin, and Dunn indices provided a balanced view of both cohesion and separation, guiding the selection of the optimal configuration for dataset construction. Although some overlap remains among transitional user groups, the overall results demonstrate that the hybrid model produces stable and interpretable clusters that serve as reliable behavioral embeddings. These embeddings form the foundation of the enriched dataset, ensuring that anomaly detection experiments incorporate context-aware profiles rather than relying solely on low-level traffic parameters.

4.2. Dataset Enrichment

The enrichment stage aims to align traffic-based features from UNSW-NB15 with behavioral embeddings extracted from Charles Proxy logs, creating a joint representation space where both sources contribute complementary information. To assess whether this cross-domain alignment is effective, we rely on representation-level metrics and visualization techniques. Traditional clustering indices are less informative here, since the goal is not to form static user groups but to evaluate how well the adversarial training integrates heterogeneous domains while preserving their internal structures. Therefore, t-distributed Stochastic Neighbor Embedding (t-SNE) is employed to project high-dimensional embeddings into two dimensions, making latent patterns interpretable. This technique is widely used to verify whether embeddings preserve neighborhood relations, expose separable clusters, and reduce domain overlap. In our context, t-SNE allows us to (i) confirm that UNSW traffic and Charles behavioral data remain distinguishable yet aligned in a shared latent space, (ii) inspect whether enrichment tightens the geometry of normal traffic patterns and refines the separation of anomalous samples, and (iii) visually validate that the learned embeddings do not collapse into indistinguishable clusters, which would indicate poor domain alignment. In short, these metrics and visualizations are necessary to ensure that enrichment produces meaningful latent representations rather than simply concatenating features without structural coherence.

To interpret how the enrichment reshapes the latent space, we apply t-distributed Stochastic Neighbor Embedding (t-SNE) to the learned embeddings [58]. t-SNE projects high-dimensional representations into two dimensions while preserving local neighborhood structure, making cluster formation, overlap, and domain separation visually apparent. This visualization allows us to (i) verify the domain separation achieved by the adversarial alignment between network traffic datasets (UNSW-NB15 and CIC-IDS-2017) and Charles behavioral data and (ii) inspect how the enrichment modifies the geometry of network embeddings by tightening normal patterns and exposing finer-grained anomalous structures. In other words, the plots provide an intuitive view of how behavioral context changes the arrangement of samples in the latent space, clarifying the mechanism by which the enriched features improve separability for downstream anomaly detection.

Figure 3 presents the t-SNE embeddings obtained from the UNSW-NB15 and Charles datasets. In the left panel, the UNSW embeddings (blue) form several clusters that correspond to distinct categories of network traffic, including normal activity and multiple attack types. The Charles embeddings (red) appear clearly separated from the network traffic, with one dominant cluster representing regular user behavior and scattered points indicating atypical or anomalous activities. This separation illustrates that the learned representations are able to capture differences not only across datasets but also within each domain, distinguishing between baseline and anomalous patterns.

The right panel shows the enriched embeddings for a subset of 100 UNSW samples. The green points form a compact cluster associated with normal traffic, while other clusters capture different categories of malicious activity. The presence of scattered points indicates

events that do not conform to the main groups, which may correspond to rare attack signatures. Compared to the original embeddings, this enriched representation yields a clearer separation of attack-related patterns, suggesting that the model captures structural differences in network behavior with greater precision.

Together, both panels indicate that the training process achieved a domain separation between network traffic and user behavior while also preserving the internal structure within each domain. This allows the representation to support the identification of normal, suspicious, and anomalous activities, which is essential for subsequent profiling and detection tasks.

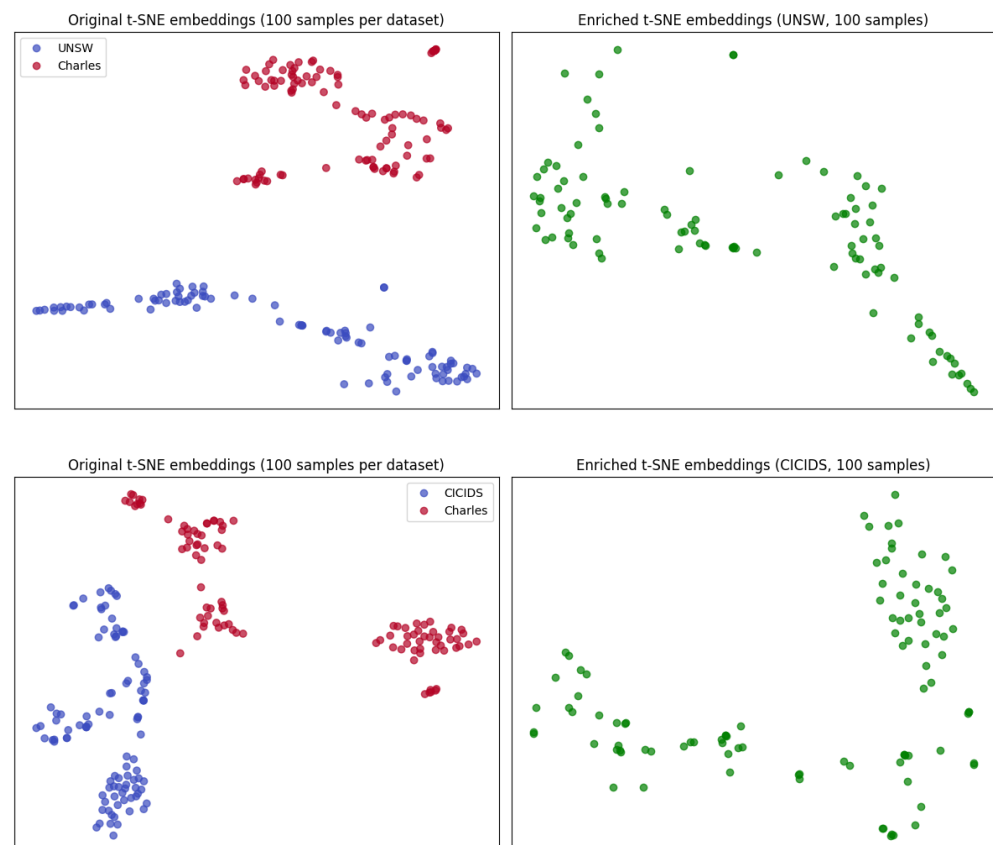


Figure 3. t-SNE visualization of learned embeddings showing domain-specific representations. (**Top row:**) Original embeddings from 100 samples each of UNSW-NB15 network traffic (blue) and Charles user behavioral data (red), and enriched UNSW-NB15 embeddings (green). (**Bottom row:**) Corresponding visualization for CIC-IDS-2017 traffic and enriched embeddings. In both cases, original traffic and behavioral data exhibit clear domain separation, while enriched representations form refined clusters that better discriminate between normal, suspicious, and anomalous network traffic patterns.

Overall, the enrichment analysis demonstrates that adversarial training successfully integrates user behavior with network traffic features while maintaining the internal structure of each domain. The clear separation between behavioral and traffic embeddings, together with the refined clustering observed in enriched datasets representations, confirms that cross-domain alignment provides a richer latent space for anomaly detection. Normal traffic becomes more compact, anomalous activities gain sharper boundaries, and rare events emerge as distinct outliers, suggesting that the added behavioral context improves the discriminative capacity of the feature space. These findings validate the necessity of the enrichment step; rather than replacing traffic parameters, behavioral embeddings act

as a regularization mechanism that enhances the separability of anomalies, laying the foundation for more robust classification in the next stage.

4.3. Anomaly Classification

The anomaly classification stage evaluates the effectiveness of integrating behavioral embeddings into traffic-based models by comparing them with a traffic-only baseline. In this context, the choice of performance metrics is critical, as different indicators capture different aspects of intrusion detection quality. Accuracy measures the overall proportion of correctly classified samples but can be misleading when attack classes are imbalanced. Precision reflects the ability of the model to correctly identify attacks among all traffic labeled as malicious, minimizing false alarms in operational scenarios. Recall quantifies the ability to detect actual attacks, which is crucial to avoid undetected intrusions. The F1-score balances precision and recall, providing a single measure that accounts for both detection capability and false positives. Finally, the ROC-AUC evaluates the model's discrimination power across thresholds, offering a robust view of performance under varying decision criteria. Together, these metrics provide a comprehensive evaluation framework that goes beyond raw accuracy, ensuring that improvements with behavioral enrichment are meaningful for real-world IDS deployment.

This section evaluates anomaly detection performance by comparing two experimental setups across both UNSW-NB15 and CIC-IDS2017 datasets: a baseline model trained on balanced datasets without enrichment and a model trained on enriched datasets that integrate behavioral profiles through adversarial autoencoder alignment. For both datasets, we sample 30,000 instances from each class (attack and normal traffic) to create balanced training sets of 60,000 samples in total. This configuration ensures fair comparison between baseline and enriched approaches while maintaining consistency across benchmarks.

The evaluation includes learning curve analysis, confusion matrices, and cross-validation metrics to provide a comprehensive assessment of classification performance. The confusion matrices highlight the distribution of false positives and false negatives in each setup, while cross-validation ensures that the results are stable across multiple folds and not tied to a single train-test split.

For the enriched configurations, the integration of behavioral profiles introduces additional variability that avoids perfect memorization. This prevents the classifier from converging to trivial solutions and promotes the learning of more robust decision boundaries. It is important to note that this benefit is specific to the controlled experimental configuration presented here; no model can completely eliminate the risk of overfitting in all scenarios. Nevertheless, the enriched datasets help the classifier balance accuracy and generalization, achieving realistic performance that is more representative of conditions likely to be encountered in deployment.

Our contribution demonstrates the technical feasibility of cross-domain alignment using adversarial autoencoders. The enrichment augments the feature space from 42 to 96 dimensions and produces consistent patterns across two benchmark datasets. We position this as a proof-of-concept for cross-domain enrichment methodologies, showing that behavioral context can be systematically integrated with network traffic features to enhance model robustness.

4.3.1. UNSW-NB15 Results

Figure 4 illustrates the training dynamics for both experimental setups on UNSW-NB15. The model trained on enriched data (Figure 4a) demonstrates gradual loss reduction over 50 epochs, with training loss decreasing from 0.67 to 0.04 and validation loss stabilizing at 0.05. The convergence pattern shows consistent decrease without abrupt changes,

maintaining a small gap between training and validation losses throughout the process. The model trained on balanced original data (Figure 4b) exhibits rapid convergence within the first 10 epochs. Both training and validation losses drop from initial values of 0.6 and 0.47, respectively, to near-zero values by epoch 6. The losses remain at approximately 0.001 for the remaining training period, creating parallel trajectories with minimal separation.

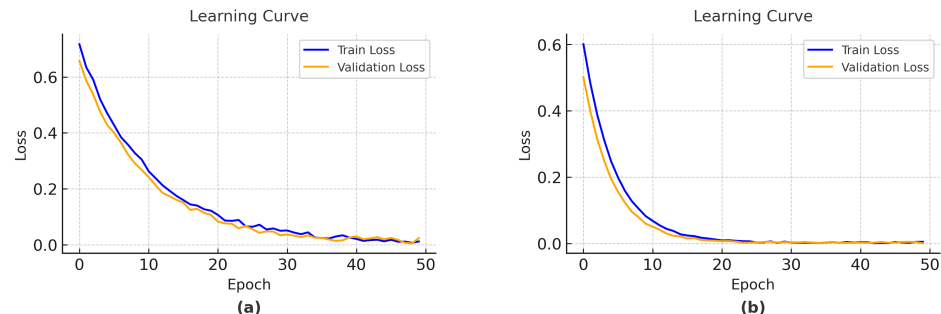


Figure 4. Learning curves comparing DFNN training dynamics on UNSW-NB15. (a) Model trained on enriched dataset shows gradual convergence over 50 epochs with training and validation losses stabilizing around 0.05, indicating controlled learning progression and maintained generalization gap. (b) Model trained on balanced original data exhibits rapid convergence by epoch 10 with both losses approaching zero, suggesting potential overfitting through dataset memorization. The sustained loss values in the enriched model demonstrate regularization effects from data augmentation, while the near-zero losses in the balanced model indicate perfect fitting to training distribution patterns.

The convergence behavior reveals fundamental differences in learning dynamics. The enriched model requires extended training periods to achieve convergence, suggesting that the augmented dataset presents a more complex optimization landscape. The sustained loss values indicate that the model continues to encounter variability in the data throughout training. The rapid convergence observed in the model without enriched data indicates that the optimization process quickly identifies patterns that allow near-perfect classification. The immediate drop to near-zero loss suggests that the model rapidly learns to exploit specific characteristics present in the training data. This behavior typically occurs when the dataset lacks sufficient complexity to challenge the model learning capacity.

The maintained gap between training and validation losses in the model with enriched data provides evidence that the data augmentation process successfully introduces regularization effects. The model's inability to achieve zero loss indicates that the enriched dataset contains sufficient variability to prevent complete memorization of training patterns. These training dynamics support the hypothesis that user behavior profiling enrichment creates learning conditions that require the model to develop more robust decision boundaries rather than relying on dataset-specific artifacts.

Figure 5 presents the confusion matrices for both experimental setups on UNSW-NB15. The model trained on enriched data (Figure 5a) produces 53 false positives (normal traffic classified as attacks) and 28 false negatives (attacks classified as normal traffic). This results in a false positive rate of 0.87% and a false negative rate of 0.47%. The error distribution demonstrates that the model maintains discrimination capability while exhibiting controlled misclassification patterns. In contrast, the model trained on original data (Figure 5b) achieves zero misclassifications across all categories. The absence of any false positives or false negatives indicates that the model achieves perfect separation between classes during validation.

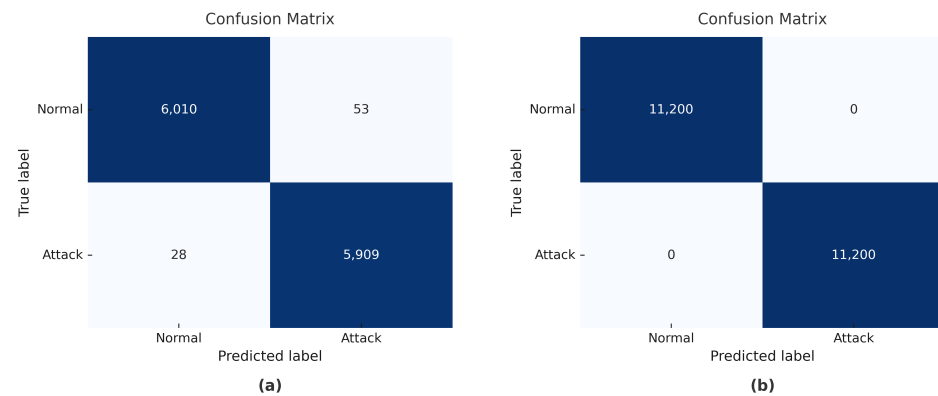


Figure 5. Confusion matrices comparing DFNN performance on UNSW-NB15. (a) Model trained on enriched dataset with user behavior profiling shows realistic classification performance with 53 false positives and 28 false negatives, achieving balanced error distribution and demonstrating robust generalization capabilities. (b) Model trained on balanced original data exhibits perfect classification with zero misclassifications, indicating potential overfitting and memorization rather than genuine learning. The enriched model's moderate error rates suggest better adaptability to real-world deployment scenarios, where perfect accuracy is unrealistic and model robustness is prioritized over training set performance. Color intensity represents the number of classifications, with darker shades indicating higher counts.

The presence of misclassification errors in the enriched model suggests that the data augmentation process introduces variability that prevents the model from achieving perfect memorization of the training patterns. The balanced error distribution between false positives and false negatives indicates that the model does not exhibit bias toward either class. The zero-error performance of the model without enriched data, while appearing optimal, raises concerns about the ability of the model to generalize beyond the specific data distribution encountered during training. Perfect classification performance on validation data typically indicates that the model has learned to exploit specific patterns or artifacts present in the dataset rather than developing robust decision boundaries.

The model trained on the enriched UNSW-NB15 dataset achieved accuracy of 97.17%, precision of 96.95%, recall of 97.34%, F1-score of 97.14%, and ROC-AUC of 99.77%. The training process converged over 31 epochs, with a small gap between training and validation losses. The model trained solely on the balanced original data reached 100% across all measures, with rapid convergence and validation loss approaching zero.

The enriched model demonstrates stable and consistently high performance across all folds, as evidenced by the results in Table 6. Accuracy, precision, recall, F1-score, and ROC-AUC remain stable, with minimal variation, indicating that the model generalizes well to unseen data and that the high performance is not caused by overfitting.

Table 6. Cross-validation metrics for the Deep Feedforward Neural Network classifier trained on the enriched UNSW-NB15 dataset. Metrics are reported for each fold and the average across folds.

Fold	Accuracy	Precision	Recall	F1-Score	ROC-AUC
1	0.9975	0.9956	0.9999	0.9977	0.9999
2	0.9980	0.9971	0.9992	0.9982	0.9998
3	0.9980	0.9968	0.9996	0.9982	0.9997
4	0.9976	0.9958	0.9998	0.9978	0.9997
5	0.9982	0.9969	0.9999	0.9984	0.9996
Average	0.9979	0.9965	0.9997	0.9981	0.9997

4.3.2. CIC-IDS2017 Results

Figure 6 illustrates the training dynamics for both experimental setups on CIC-IDS2017. The model trained on enriched data (Figure 6a) demonstrates a similar gradual convergence pattern, with training and validation losses decreasing progressively over the training period. Both curves converge towards low loss values while maintaining a small but consistent gap, indicating controlled learning without memorization. The model trained on balanced original CIC-IDS2017 data (Figure 6b) exhibits rapid convergence comparable to the UNSW-NB15 baseline, with both training and validation losses dropping sharply in the initial epochs and stabilizing at near-zero values. This pattern mirrors the behavior observed in the UNSW-NB15 baseline, suggesting consistent overfitting characteristics across datasets when behavioral enrichment is absent.

The consistency of convergence patterns across both datasets provides evidence that behavioral enrichment produces systematic regularization effects independent of the specific network traffic characteristics. The enriched models in both UNSW-NB15 and CIC-IDS2017 require extended training periods and maintain validation gaps, while both baselines converge rapidly to near-perfect training performance.

Figure 7 presents the confusion matrices for both experimental setups on CIC-IDS2017. The model trained on enriched data (Figure 7a) produces 4 false positives and 4 false negatives, resulting in a false positive rate of 0.02% and a false negative rate of 0.07%. This balanced error distribution, while lower in absolute numbers compared to UNSW-NB15, maintains the characteristic pattern of controlled misclassification that indicates robust generalization. In contrast, the model trained on original CIC-IDS2017 data (Figure 7b) achieves zero misclassifications across all categories, mirroring the perfect separation observed in the UNSW-NB15 baseline.

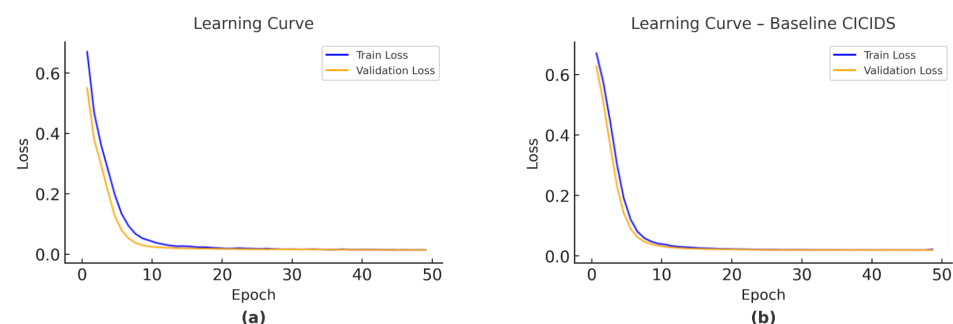


Figure 6. Learning curves comparing DFNN training dynamics on CIC-IDS2017. (a) Model trained on enriched dataset shows gradual convergence with training and validation losses maintaining a controlled gap, demonstrating regularization effects from behavioral enrichment. (b) Model trained on balanced original data exhibits rapid convergence with near-zero losses, indicating potential overfitting, consistent with patterns observed in UNSW-NB15. The sustained learning progression in the enriched model contrasts with the rapid memorization behavior of the baseline.

The model trained on the enriched CIC-IDS2017 dataset achieved accuracy of 99.93%, precision of 99.93%, recall of 99.93%, F1-score of 99.93%, and ROC-AUC of 99.99%. While these metrics are higher than those observed with UNSW-NB15 enrichment (97%), they remain below the perfect 100% baseline performance, maintaining the characteristic gap that indicates generalization capability. The model trained solely on the balanced original CIC-IDS2017 data reached 100% across all measures, consistent with the baseline behavior observed in UNSW-NB15.

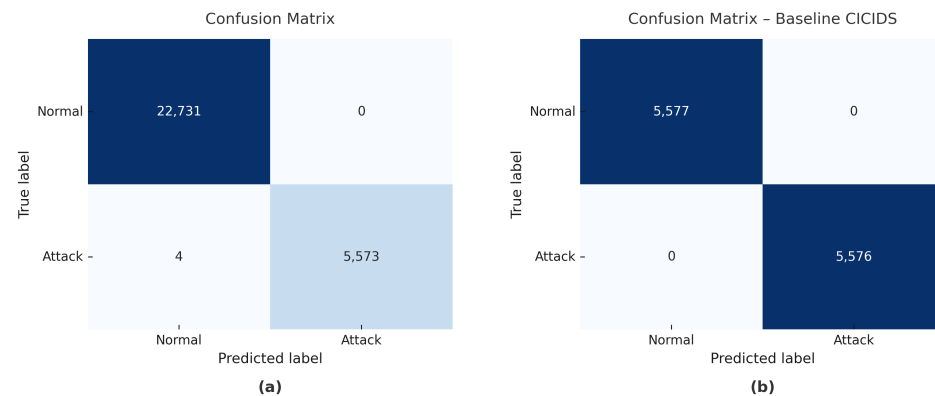


Figure 7. Confusion matrices comparing DFNN performance on CIC-IDS2017. **(a)** Model trained on enriched dataset shows balanced error distribution with four false positives and four false negatives, demonstrating controlled generalization. **(b)** Model trained on balanced original data exhibits perfect classification with zero misclassifications, consistent with memorization patterns observed across datasets. The enriched model maintains realistic error rates that suggest adaptability to deployment conditions.

The enriched CIC-IDS2017 model demonstrates remarkably stable performance across all folds, as shown in Table 7. The minimal variation across folds (all metrics at 99.93%) indicates consistent generalization behavior. The slightly higher absolute performance compared to UNSW-NB15 enrichment may reflect differences in traffic patterns or attack distributions between datasets, but the consistent presence of controlled error rates across both benchmarks confirms the systematic mechanism to avoid the perfect memorization effect of behavioral enrichment.

Table 7. Cross-validation metrics for the Deep Feedforward Neural Network classifier trained on the enriched CIC-IDS2017 dataset. Metrics are reported for each fold and the average across folds.

Fold	Accuracy	Precision	Recall	F1-Score	ROC-AUC
1	0.9993	0.9993	0.9993	0.9993	0.9999
2	0.9993	0.9993	0.9993	0.9993	0.9999
3	0.9994	0.9993	0.9994	0.9993	0.9999
4	0.9993	0.9993	0.9993	0.9993	0.9999
5	0.9993	0.9993	0.9993	0.9993	0.9999
Average	0.9993	0.9993	0.9993	0.9993	0.9999

The observed behavior across both datasets highlights the value of improving the representational richness of the input space. The enriched datasets increase variability and complexity without requiring additional raw data, demonstrating that robust generalization can be achieved through data enrichment rather than architectural modifications. This effect is particularly relevant because it shows that simpler models can reach levels of reliability that would typically require greater capacity or significantly larger datasets when provided with behaviorally enriched features.

While the baseline models achieve perfect validation performance on both balanced datasets, such results are atypical in real-world deployment where data distributions shift, attack variants emerge, and computational constraints favor simpler models. Our enrichment framework demonstrates that behavioral context enables high performance (97%+ for UNSW-NB15, 99.9%+ for CIC-IDS2017) with enhanced robustness, as evidenced by (1) stable cross-validation across folds in both datasets, (2) balanced error distribution preventing bias toward false negatives or positives, and (3) training dynamics that consistently suggest learning generalizable patterns rather than dataset-specific artifacts.

The comparison across two benchmark datasets indicates that data enrichment with user behavior profiling acts as a systematic regularization mechanism, even with simple neural network architectures. By introducing variability through behavioral context derived from adversarial autoencoder alignment, the enriched datasets maintain the complexity of network traffic while preventing the model from memorizing training patterns. The consistency of this effect across UNSW-NB15 and CIC-IDS2017—despite their different traffic characteristics and attack distributions—validates the technical feasibility of cross-domain alignment for intrusion detection.

The performance difference between enriched models (97–99.9%) and baseline models (100%) should not be interpreted as the baseline being superior. Rather, the baseline's perfect validation scores under balanced conditions represent an upper bound achievable through dataset-specific optimization. The enriched models' slightly lower but highly stable performances across folds, combined with training dynamics showing sustained variability in both datasets, indicates learning patterns that may generalize better to distribution shifts encountered in deployment. This tradeoff between validation perfection and operational robustness aligns with established principles in machine learning regularization.

These results support the use of cross-domain behavioral enrichment with network traffic data to enhance robustness across architectural designs. The enrichment framework creates training scenarios that prepare models for deployment in network environments where computational efficiency and model interpretability are priorities. The consistency of the regularization effects across two distinct benchmark datasets demonstrates that adversarial autoencoder-based alignment can systematically integrate behavioral context with traffic features, producing models that are less brittle and more reliable for deployment in dynamic network environments.

5. Conclusions and Future Work

This research developed an anomaly detection system that integrates user profiling with behavioral enrichment through cross-domain latent alignment. The system operates in three stages: clustering user behavior via autoencoder embeddings, enhancing behavior through adversarial autoencoders, and classifying anomalies with a deep feedforward neural network.

The hybrid approach to user profiling combines autoencoders with agglomerative clustering. Clustering on raw traffic data alone produces moderate or inconsistent results, as the original feature space does not fully separate user behaviors. Reducing dimensionality with the autoencoder emphasizes essential patterns and reduces noise, enabling the clustering algorithm to identify coherent behavioral structures. As a result, the hybrid approach improves cluster consistency and stability across different configurations, demonstrating that the observed gains arise from exposing underlying patterns not directly visible in the raw data.

Behavioral enrichment aligns latent representations from the UNSW-NB15 and CIC-IDS2017 datasets and user profiling embeddings using adversarial autoencoders. The discriminator distinguishes domain-specific patterns while maintaining internal consistency, and concatenating the average profile embeddings with network traffic representations reduces variance in the latent space. This integration provides global behavioral context, which supplements traffic features and contributes to more structured and informative representations for anomaly detection. The main performance improvements are attributable to this cross-domain alignment, which preserves classification accuracy and supports the identification of anomalous patterns without degrading performance.

The deep feedforward neural network trained on the enriched feature space converges rapidly and exhibits stable learning, indicating that the combined representation provides

coherent signals for classification. Confusion matrix analysis shows low rates of false positives and false negatives, suggesting that the model can effectively distinguish normal from anomalous traffic. These results confirm that the system can combine heterogeneous data sources in a technically feasible manner while maintaining operational performance.

The experiments show that behavioral enrichment modifies training dynamics. While the traffic-only baseline achieves perfect validation accuracy on the balanced UNSW-NB15 dataset, this performance largely reflects memorization. In contrast, the enriched model maintains high detection rates (97%+ across metrics) but with more gradual convergence, stable cross-validation results, and balanced error distributions between false positives and negatives. These patterns suggest that cross-domain enrichment introduces controlled variability that acts as a regularization mechanism, helping the model learn more generalizable decision boundaries. In practical terms, behavioral embeddings complement traditional traffic parameters, such as packet sizes, intervals, and protocols, by providing additional contextual information. This enrichment preserves discriminative power while enhancing resilience to overfitting, indicating that the approach can strengthen anomaly detection beyond traffic-only baselines, though broader validation across datasets and deployment scenarios remains necessary.

The hybrid clustering approach enhances user profiling by revealing latent structures in network traffic data. Behavioral enrichment through adversarial alignment integrates these profiles with traffic features, producing a more informative feature space for classification. The combination of dimensionality reduction, clustering, and cross-domain alignment explains the observed performance improvements. Overall, the results demonstrate that incorporating behavioral context into network anomaly detection supports effective and robust detection, with potential for future exploration of instance-specific embeddings or additional alignment strategies.

The evaluation was conducted on a single public dataset combined with proxy user profiling data, which may limit the generalizability of the results to other network environments. Additionally, the scalability of the proposed approach to large-scale or real-time network monitoring has not been assessed. Future work could explore the application of the method across diverse datasets and evaluate its performance in operational deployments.

Future research directions should explore the scalability of the proposed system across different network environments and traffic volumes. An investigation of alternative dimensionality reduction techniques beyond autoencoders could reveal more efficient feature extraction methods for user profiling. The extension of the adversarial training framework to incorporate multiple data sources beyond network traffic and user profiles could enhance detection capabilities for complex attack scenarios. Real-time implementation studies would validate the practical deployment feasibility and identify performance bottlenecks in operational environments. Research into explainable AI techniques for the proposed model could improve interpretability of anomaly detection decisions, facilitating security analyst understanding and system trust. The investigation of federated learning approaches could enable collaborative anomaly detection across multiple network domains while preserving data privacy.

Author Contributions: Conceptualization, C.S.-N. and L.J.D.P.; methodology, C.S.-N. and L.J.D.P.; software, C.S.-N. and L.J.D.P.; validation, C.S.-N. and L.J.D.P.; formal analysis, C.S.-N. and L.J.D.P.; investigation, C.S.-N. and L.J.D.P.; resources, C.S.-N. and L.J.D.P.; data curation, C.S.-N. and L.J.D.P.; writing—original draft preparation, C.S.-N. and L.J.D.P.; writing—review and editing, C.S.-N. and L.J.D.P.; visualization, C.S.-N. and L.J.D.P.; supervision, C.S.-N. and L.J.D.P.; project administration, C.S.-N. and L.J.D.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding authors.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Sen, J.; Dasgupta, S. Data Privacy Preservation on the Internet of Things. *arXiv* **2023**, arXiv:2304.00258. [\[CrossRef\]](#)
2. Hategekimana, J.B. Growth of Internet Users with Special Emphasis on the Impact of New Coronary Pneumonia. *J. Soc. Sci. Humanit.* **2024**, *6*, 84–93. [\[CrossRef\]](#) [\[PubMed\]](#)
3. Peng, P. Research on Computer Network Security Vulnerabilities and Encryption Technology in Cloud Computing Environment. *Appl. Math. Nonlinear Sci.* **2024**, *9*, 171. [\[CrossRef\]](#)
4. Yu, Y. Big data network security defense mode of deep learning algorithm. *Open Comput. Sci.* **2022**, *12*, 345–356. [\[CrossRef\]](#)
5. Xi, Z.; Zhou, Y.; Zhang, D.; Gao, K.; Sun, C.; Cao, J.; Wang, Y.; Xu, M.; Wu, J. Newton: Intent-Driven Network Traffic Monitoring. *IEEE/ACM Trans. Netw.* **2022**, *30*, 939–952. [\[CrossRef\]](#)
6. Trevisan, M.; Giordano, D.; Drago, I.; Munafò, M.M.; Mellia, M. Five Years at the Edge: Watching Internet From the ISP Network. *IEEE/ACM Trans. Netw.* **2020**, *28*, 561–574. [\[CrossRef\]](#)
7. Wang, S.; Balarezo, J.F.; Kandeepan, S.; Al-Hourani, A.; Chavez, K.G.; Rubinstein, B. Machine learning in network anomaly detection: A survey. *IEEE Access* **2021**, *9*, 152379–152396. [\[CrossRef\]](#)
8. Hooshmand, M.K.; Hosahalli, D. Network anomaly detection using deep learning techniques. *CAAI Trans. Intell. Technol.* **2022**, *7*, 228–243. [\[CrossRef\]](#)
9. Al-Meer, A.; Al-Kuwari, S. Physical Unclonable Functions (PUF) for IoT Devices. *ACM Comput. Surv.* **2022**, *55*, 1–31. [\[CrossRef\]](#)
10. Zhong, D.; Liu, J.; Xiao, M.; Xie, Y.; Shi, H.; Liu, L.; Zhao, C.; Ding, L.; Peng, L.; Zhang, Z. Twin physically unclonable functions based on aligned carbon nanotube arrays. *Nat. Electron.* **2022**, *5*, 424–432. [\[CrossRef\]](#)
11. Chen, P.; Li, D.; Li, Z.; Xu, X.; Wang, H.; Zhou, X.; Zhai, T. Programmable Physical Unclonable Functions Using Randomly Anisotropic Two-Dimensional Flakes. *ACS Nano* **2023**, *17*, 23989–23997. [\[CrossRef\]](#)
12. Marri, S.; K, A. A Study on FPGA Implementation of Physical Unclonable Functions (PUFs). *Int. J. Innov. Res. Comput. Commun. Eng.* **2024**, *12*, 5–13. [\[CrossRef\]](#)
13. Albright, A.; Gelfand, B.; Dixon, M. Learnability of Optical Physical Unclonable Functions Through the Lens of Learning with Errors. *IEEE Trans. Inf. Forensics Secur.* **2025**, *20*, 886–897. [\[CrossRef\]](#)
14. Duan, X.; Fu, Y.; Wang, K. Network traffic anomaly detection method based on multi-scale residual classifier. *Comput. Commun.* **2023**, *198*, 206–216. [\[CrossRef\]](#)
15. Aouedi, O.; Le, V.A.; Piamrat, K.; Ji, Y. Deep learning on network traffic prediction: Recent advances, analysis, and future directions. *ACM Comput. Surv.* **2025**, *57*, 1–37. [\[CrossRef\]](#)
16. Kamal, H.; Mashaly, M. Enhanced Hybrid Deep Learning Models-Based Anomaly Detection Method for Two-Stage Binary and Multi-Class Classification of Attacks in Intrusion Detection Systems. *Algorithms* **2025**, *18*, 69. [\[CrossRef\]](#)
17. Elsayed, S.; Mohamed, K.; Madkour, M.A. A comparative study of using deep learning algorithms in network intrusion detection. *IEEE Access* **2024**, *12*, 58851–58870. [\[CrossRef\]](#)
18. Hooshmand, M.K.; Huchaiah, M.D.; Alzighaibi, A.R.; Hashim, H.; Atlam, E.S.; Gad, I. Robust network anomaly detection using ensemble learning approach and explainable artificial intelligence (XAI). *Alex. Eng. J.* **2024**, *94*, 120–130. [\[CrossRef\]](#)
19. Marfo, W.; Tosh, D.K.; Moore, S.V. Enhancing network anomaly detection using graph neural networks. In Proceedings of the 2024 22nd Mediterranean Communication and Computer Networking Conference (MedComNet), Nice, France, 11–13 June 2024; pp. 1–10.
20. Altulaihan, E.; Almaiah, M.A.; Aljughaiman, A. Anomaly detection IDS for detecting DoS attacks in IoT networks based on machine learning algorithms. *Sensors* **2024**, *24*, 713. [\[CrossRef\]](#)
21. Kalnoor, G.; Gowrishankar, S. IoT-based smart environment using intelligent intrusion detection system. *Soft Comput.* **2021**, *25*, 11573–11588. [\[CrossRef\]](#)
22. Rani, D.; Gill, N.; Gulia, P.; Arena, F.; Pau, G. Design of an intrusion detection model for IoT-enabled smart home. *IEEE Access* **2023**, *11*, 52509–52526. [\[CrossRef\]](#)
23. Murthy, A.; Asghar, M.R.; Tu, W. A lightweight Intrusion Detection for Internet of Things-based smart buildings. *Security Priv.* **2024**, *7*, e386. [\[CrossRef\]](#)
24. Javed, A.; Awais, M.; Qureshi, A.; Jawad, M.; Arshad, J.; Larijani, H. Embedding tree-based intrusion detection system in smart thermostats for enhanced IoT security. *Sensors* **2024**, *24*, 27320. [\[CrossRef\]](#) [\[PubMed\]](#)
25. Wang, M.; Yang, N.; Weng, N. Securing a smart home with a transformer-based IoT intrusion detection system. *Electronics* **2023**, *12*, 2100. [\[CrossRef\]](#)

26. Garroppo, R.; Giardina, P.; Landi, G.; Ruta, M. Trustworthy AI and federated learning for intrusion detection in 6G-connected smart buildings. *Future Internet* **2025**, *17*, 191. [\[CrossRef\]](#)
27. Oran, S.; Koçak, A. Security review and performance analysis of QUIC and TCP protocols. In Proceedings of the 15th International Conference on Information Security and Cryptography (ISCTURKEY), Ankara, Turkey, 19–20 October 2022; p. 9931821. [\[CrossRef\]](#)
28. Almuhammadi, S.; Al-Bakhat, L. Intrusion detection on QUIC traffic: A machine learning approach. In Proceedings of the 2022 7th International Conference on Data Science and Machine Learning Applications (CDMA), Riyadh, Saudi Arabia, 1–3 March 2022.
29. Benova, L.; Hudec, L. Comprehensive Analysis and Evaluation of Anomalous User Activity in Web Server Logs. *Sensors* **2024**, *24*, 746. [\[CrossRef\]](#)
30. Li, Z.; Rios, A.L.G.; Trajković, L. Machine Learning for Detecting Anomalies and Intrusions in Communication Networks. *IEEE J. Sel. Areas Commun.* **2021**, *39*, 2254–2264. [\[CrossRef\]](#)
31. Modell, A.; Larson, J.; Turcotte, M.J.M.; Bertiger, A. A Graph Embedding Approach to User Behavior Anomaly Detection. In Proceedings of the 2021 IEEE International Conference on Big Data (Big Data), Orlando, FL, USA, 15–18 December 2021; pp. 2650–2655. [\[CrossRef\]](#)
32. Martín, A.G.; Fernández-Isabel, A.; Martín de Diego, I.; Beltrán, M. A survey for user behavior analysis based on machine learning techniques: Current models and applications. *Appl. Intell.* **2021**, *51*, 6029–6055. [\[CrossRef\]](#)
33. Tun, M.T.; Nyaung, D.E.; Phyu, M.P. Network anomaly detection using threshold-based sparse. In Proceedings of the 11th International Conference on Advances in Information Technology, Bangkok, Thailand, 1–3 July 2020; pp. 1–8.
34. Pandhurnekar, V.; Iyyappan, A.; Dhok, D.; Khante, V.; Wazalwar, S.S. Proposed Method for Threat Detection Using User Behavior Analysis. In Proceedings of the 2023 IEEE 3rd International Conference on Technology, Engineering, Management for Societal impact using Marketing, Entrepreneurship and Talent (TEMSMET), Mysuru, India, 10–11 February 2023; pp. 1–5. [\[CrossRef\]](#)
35. Alshehri, A.; Khan, N.; Alowayr, A.; Alghamdi, M.Y. Cyberattack Detection Framework Using Machine Learning and User Behavior Analytics. *Comput. Syst. Sci. Eng.* **2023**, *44*, 1679–1689. [\[CrossRef\]](#)
36. He, J.; Yin, X. Internet User Behavior Analysis Based on Big Data. In Proceedings of the 2021 International Wireless Communications and Mobile Computing (IWCMC), Harbin, China, 28 June–2 July 2021; pp. 432–435. [\[CrossRef\]](#)
37. Wang, L.; Jones, R. Big Data Analytics in Cyber Security: Network Traffic and Attacks. *J. Comput. Inf. Syst.* **2020**, *61*, 410–417. [\[CrossRef\]](#)
38. Tao, X.; Lu, S.; Zhao, F.; Lan, R.; Chen, L.; Fu, L.; Jia, R. User Behavior Threat Detection Based on Adaptive Sliding Window GAN. *IEEE Trans. Netw. Serv. Manag.* **2024**, *21*, 2493–2503. [\[CrossRef\]](#)
39. Moustafa, N.; Slay, J. UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In Proceedings of the 2015 Military Communications and Information Systems Conference (MilCIS), Canberra, Australia, 10–12 November 2015; pp. 1–6.
40. Rosay, A.; Cheval, E.; Carlier, F.; Leroux, P. Network intrusion detection: A comprehensive analysis of CIC-IDS2017. In *Proceedings of the 8th International Conference on Information Systems Security and Privacy, Online, 8–10 February 2022*; SCITEPRESS-Science and Technology Publications: Setubal, Portugal, 2022; pp. 25–36.
41. Li, Z.; Chen, X.; Song, J.; Gao, J. Adaptive label propagation for group anomaly detection in large-scale networks. *IEEE Trans. Knowl. Data Eng.* **2023**, *35*, 12053–12067. [\[CrossRef\]](#)
42. P, S.; Palaniappan, K.; Duraipandi, B.; Balasubramanian, U. Dynamic behavioral profiling for anomaly detection in software-defined IoT networks: A machine learning approach. *Peer-to-Peer Netw. Appl.* **2024**, *17*, 2450–2469. [\[CrossRef\]](#)
43. Shafi, M.; Lashkari, A.H.; Roudsari, A.H. NTLFlowLyzer: Towards generating an intrusion detection dataset and intruders behavior profiling through network and transport layers traffic analysis and pattern extraction. *Comput. Secur.* **2024**, *148*, 104160. [\[CrossRef\]](#)
44. Singh, I.; Jindal, R. Outlier based intrusion detection in databases for user behaviour analysis using weighted sequential pattern mining. *Int. J. Mach. Learn. Cybern.* **2023**, *15*, 2573–2593. [\[CrossRef\]](#)
45. Abiramasundari, S.; Ramaswamy, V. Distributed denial-of-service (DDOS) attack detection using supervised machine learning algorithms. *Sci. Rep.* **2025**, *15*, 13098. [\[CrossRef\]](#)
46. Bashurov, V.; Safonov, P. Anomaly detection in network traffic using entropy-based methods: Application to various types of cyberattacks. *Issues Inf. Syst.* **2023**, *24*, 82–94.
47. Zhao, Z.; Guo, H.; Wang, Y. A multi-information fusion anomaly detection model based on convolutional neural networks and AutoEncoder. *Sci. Rep.* **2024**, *14*, 16147. [\[CrossRef\]](#)
48. Kasongo, S.M.; Sun, Y. Performance analysis of intrusion detection systems using a feature selection method on the UNSW-NB15 dataset. *J. Big Data* **2020**, *7*, 105. [\[CrossRef\]](#)
49. Greenacre, M.; Groenen, P.J.; Hastie, T.; d’Enza, A.I.; Markos, A.; Tuzhilina, E. Principal component analysis. *Nat. Rev. Methods Prim.* **2022**, *2*, 100. [\[CrossRef\]](#)
50. Aggarwal, C.C. *Recommender Systems: The Textbook*; Springer: Berlin/Heidelberg, Germany, 2016.

51. Cover, T.M.; Thomas, J.A. *Elements of Information Theory*; John Wiley & Sons: Hoboken, NJ, USA, 2006.
52. Ikotun, A.M.; Ezugwu, A.E.; Abualigah, L.; Abuhaija, B.; Heming, J. K-means clustering algorithms: A comprehensive review, variants analysis, and advances in the era of big data. *Inf. Sci.* **2023**, *622*, 178–210. [[CrossRef](#)]
53. Tokuda, E.K.; Comin, C.H.; Costa, L.d.F. Revisiting agglomerative clustering. *Phys. A Stat. Mech. Its Appl.* **2022**, *585*, 126433. [[CrossRef](#)]
54. Singh, H.V.; Girdhar, A.; Dahiya, S. A Literature survey based on DBSCAN algorithms. In Proceedings of the 2022 6th International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, 25–27 May 2022; pp. 751–758.
55. Shahapure, K.R.; Nicholas, C. Cluster quality analysis using silhouette score. In Proceedings of the 2020 IEEE 7th international conference on data science and advanced analytics (DSAA), Sydney, Australia, 6–9 October 2020; pp. 747–748.
56. Muntean, M.; Militaru, F.D. Metrics for evaluating classification algorithms. In *Proceedings of the Education, Research and Business Technologies: Proceedings of 21st International Conference on Informatics in Economy (IE 2022)*, Bucharest, Romania, 26–27 May 2022; Springer: Berlin/Heidelberg, Germany, 2023; pp. 307–317.
57. Shirdel, M.; Di Mauro, M.; Liotta, A. Worthiness Benchmark: A novel concept for analyzing binary classification evaluation metrics. *Inf. Sci.* **2024**, *678*, 120882. [[CrossRef](#)]
58. Marx, V. Seeing data as t-SNE and UMAP do. *Nat. Methods* **2024**, *21*, 930–933. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.