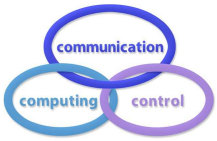
INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
Online ISSN 1841-9844, ISSN-L 1841-9836, Volume: 20, Issue: 4, Month: August, Year: 2025
Article Number: 6890, https://doi.org/10.15837/ijccc.2025.4.6890



CCC Publications

A Normalizing Flow-Based Semi-Supervised Method for Imbalanced Network Intrusion Detection

Chaoqun Guo, Shunjie Yang, Jubao Cheng, Dalin Zhang

Chaoqun Guo[†], **Shunjie Yang**[†], **Jubao Cheng**[†]
School of Software Engineering
Beijing Jiaotong University, Beijing
No.3 Shangyuancun Haidian District Beijing 100044 P. R. China
[†] contributed equally to this work
{chqguo,shunjie,chengjubao}@bjtu.edu.cn

Dalin Zhang*
1. Intelligent Systems and Security Laboratory Of BJTU
2. School of Cyberspace Security
Beijing Jiaotong University, Beijing
No.3 Shangyuancun Haidian District Beijing 100044 P. R. China
*Corresponding author: dalin@bjtu.edu.cn

Abstract

Intrusion Detection Systems (IDS) are integral to ensuring network security. However, in practical settings, network traffic data often exhibits significant imbalances, affecting both labeled and unlabeled data distributions. Such imbalances notably degrade the performance of existing intrusion detection methods, particularly in semi-supervised learning contexts, where traditional approaches struggle to effectively leverage large amounts of unlabeled data for enhanced detection capabilities. This paper introduces a semi-supervised learning approach based on normalizing flows to mitigate the data imbalance issue in network intrusion detection. Normalizing flows construct flexible and invertible probabilistic models that can accurately capture and generate complex, high-dimensional network traffic data distributions. Specifically, this method utilizes a small amount of labeled data for initial training and incorporates manifold learning and self-training with unlabeled data to adapt the model to the imbalance in the unlabeled data distribution, thereby improving overall detection performance. Experimental results demonstrate that this method outperforms traditional approaches in addressing data imbalance in intrusion detection. The proposed method not only improves detection accuracy and recall but also significantly reduces reliance on data distribution assumptions, demonstrating robustness and generalization across diverse network traffic datasets.

Keywords: intrusion detection systems, normalizing flows, semi-supervised learning, data imbalance.

1 Introduction

Intrusion Detection Systems play a pivotal role in safeguarding network security by monitoring and analyzing network traffic to identify malicious activities or policy violations. The significance of IDS arises from the increasing complexity and volume of cyberattacks, which pose serious threats to the integrity, confidentiality, and availability of data and services in digital environments. The critical importance of intrusion detection is further underscored by the necessity to protect sensitive information, ensure service continuity, and comply with regulatory requirements. An effective IDS can detect a wide range of attacks, from basic brute force attempts to sophisticated Advanced Persistent Threats (APTs), thereby facilitating timely mitigation and response. However, despite technological advancements, the growing sophistication of attacks and the ever-increasing volume of network traffic demand continuous improvements in detection methodologies.

Intrusion detection methods are generally classified into two categories: signature-based and anomaly-based approaches [17]. Signature-based detection relies on predefined threat patterns (signatures) to identify malicious activities. While highly effective against known attacks, this method is limited in addressing zero-day exploits and requires frequent updates to the signature database. In contrast, anomaly-based detection models typical network behavior and flags deviations as potential threats. This approach is effective in identifying novel attacks but is frequently hindered by high false positive rates arising from the dynamic nature of network traffic. Both approaches have unique strengths and limitations, highlighting the necessity for adaptive and robust detection strategies.

Supervised learning methods have been extensively employed in intrusion detection, where models are trained on labeled datasets to differentiate between benign and malicious traffic. However, these methods face several challenges. Acquiring labeled data is both resource-intensive and time-consuming, as it necessitates expert knowledge. Furthermore, reliance on large amounts of labeled data restricts the scalability of supervised methods [2]. Additionally, real-world network traffic often exhibits a long-tail distribution, with benign traffic significantly outnumbering malicious instances. This imbalance can result in biased models that are ineffective at detecting rare yet critical attacks. These challenges underscore the need for alternative learning paradigms capable of leveraging both labeled and unlabeled data effectively.

To address the limitations of fully supervised methods, semi-supervised learning leverages a small amount of labeled data in combination with a large corpus of unlabeled data [43]. This approach reduces labeling costs while enhancing scalability. Semi-supervised learning is particularly advantageous in tackling data imbalance by utilizing the abundant unlabeled data to improve model robustness and generalization. By integrating both types of data, as depicted in Figure 1, semi-supervised methods achieve a balance between accuracy and efficiency. However, effectively addressing the imbalance in both labeled and unlabeled data remains a significant challenge in intrusion detection.

Data imbalance in intrusion detection can lead to several adverse outcomes. Models can become biased toward the majority class, leading to suboptimal detection rates for rare attacks. Furthermore, overreliance on limited labeled data can result in overfitting, diminishing the model's ability to generalize to novel threats. Addressing such imbalances necessitates sophisticated techniques to manage and mitigate their effects. Existing solutions [11, 18, 45] for managing data imbalance include data augmentation, resampling methods [5], and cost-sensitive learning [50]. Data augmentation entails generating synthetic samples for the minority class to balance the dataset. Resampling methods involve either oversampling the minority class or undersampling the majority class to achieve a balanced distribution. Cost-sensitive learning assigns higher misclassification costs to minority class instances, thereby biasing the model toward detecting them. While these techniques offer improvements, they often encounter limitations in scalability and effectiveness when applied to highly imbalanced datasets.

Normalizing flows [31] provide a robust framework for modeling complex data distributions through a series of invertible transformations. Their primary advantages include flexibility in capturing intricate data distributions without stringent assumptions, invertibility for exact likelihood estimation and efficient sampling, and scalability in high-dimensional spaces, making them particularly suitable for large-scale network traffic analysis [35]. These attributes position normalizing flows as a promising solution to the challenges posed by data imbalance in intrusion detection.

In the context of intrusion detection, normalizing flows enhance the detection of rare and novel

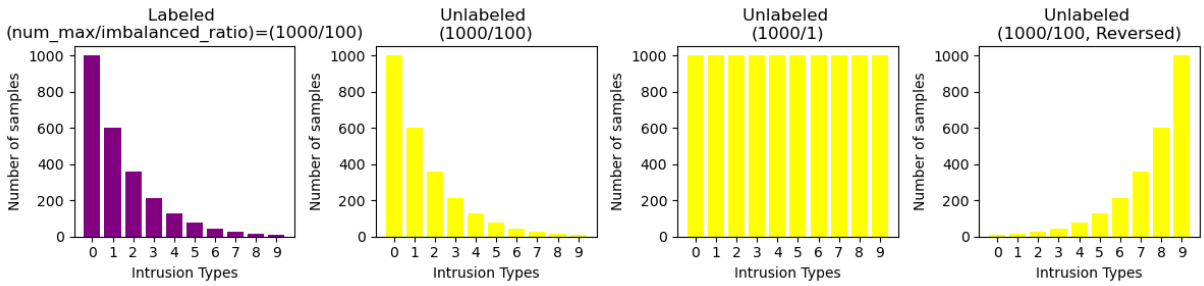


Figure 1: Distribution imbalance in labeled and unlabeled data in IDS, and exponential sampling for multi-class imbalance in this work.

attacks by precisely modeling the underlying distribution of network traffic. The integration of normalizing flows with semi-supervised learning effectively leverages both labeled and unlabeled data, thereby improving detection performance in the presence of data imbalance. This integration fosters a deeper understanding of network traffic patterns and bolsters the robustness of detection mechanisms.

This paper introduces a semi-supervised intrusion detection approach that employs normalizing flows. The primary contributions of this work are as follows:

- Introducing a novel semi-supervised model that utilizes normalizing flows to mitigate data imbalance in intrusion detection;
- Developing a self-supervised learning strategy to improve the model's adaptability to imbalanced and unlabeled data;
- Demonstrating the effectiveness and superiority of the proposed method through extensive experiments conducted on real-world network traffic datasets.

By tackling the challenges of data imbalance and capitalizing on the strengths of normalizing flows, our method provides a robust and scalable solution for enhancing intrusion detection in diverse and dynamic network environments.

2 Related Work

The landscape of intrusion detection has evolved significantly due to technological advancements and the increasing sophistication of cyber threats. To understand the current state and challenges in this field, it is crucial to examine the development of intrusion detection systems, the application of machine learning and deep learning techniques, and the emerging role of normalizing flows. This section explores these aspects to identify existing gaps and the potential solutions offered by our proposed method.

First, we examine the current state and development of intrusion detection systems. The shift from signature-based to anomaly-based methods underscores the need for more adaptive and robust approaches that can address modern cyber threats. Despite these advancements, significant challenges remain, particularly with regard to the scalability and effectiveness of these systems in real-world scenarios.

Second, we explore the application of machine learning and deep learning techniques in intrusion detection. These techniques have greatly improved detection capabilities by leveraging large datasets to identify complex patterns indicative of malicious activity. However, the reliance on labeled data and the pervasive issue of data imbalance present substantial challenges. We analyze how these issues constrain the performance of existing methods, particularly in semi-supervised settings where data imbalance can severely affect detection accuracy.

Finally, we focus on the emerging use of normalizing flows in intrusion detection. Normalizing flows offer a novel approach to modeling complex data distributions, with significant potential to overcome the limitations of traditional machine learning and deep learning methods. While promising, the use of

normalizing flows in this domain is still in its early stages, and several challenges remain. We examine how normalizing flows differ from other methods and their potential to handle imbalanced data more effectively within a semi-supervised framework.

By synthesizing insights from these three areas, we establish the foundation for our proposed method, which utilizes normalizing flows to enhance intrusion detection in the presence of data imbalance. This integrative approach aims to overcome the limitations of existing methods and improve the overall robustness and scalability of intrusion detection systems.

2.1 Intrusion Detection

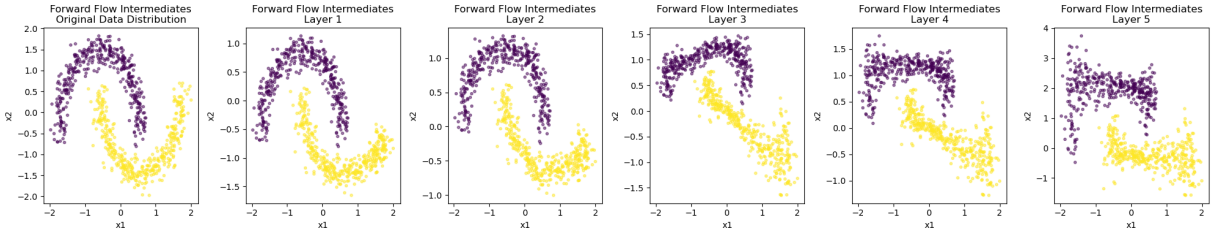
Intrusion Detection Systems have significantly evolved over the past few decades to address the growing threat landscape. Traditional IDS techniques were primarily signature-based [20, 25, 28], relying on predefined patterns of known threats. These systems [7, 32], such as Snort and Bro, detect intrusions by matching network traffic against a database of known attack signatures. Although effective at detecting known attacks, these methods struggle with new, unknown threats and require frequent updates to maintain their effectiveness. This limitation has become increasingly problematic as the volume and sophistication of cyberattacks continue to rise, creating a significant gap in the detection capabilities of traditional systems.

As the cyber threat landscape evolved, anomaly-based detection methods [48] gained prominence. Unlike signature-based approaches, anomaly-based IDS models normal network behavior and identifies deviations from this baseline as potential threats. This paradigm shift enables the detection of previously unseen attacks, which are becoming increasingly common in the modern cyber environment. Anomaly-based systems utilize statistical models, machine learning algorithms, and behavioral analysis to define what constitutes normal activity within a network. When network traffic deviates from this established norm, the system flags it as suspicious. Despite their potential, these methods often suffer from high false positive rates, where normal variations in network traffic are mistakenly identified as threats. This not only overwhelms security analysts with false alarms but also undermines trust in the IDS.

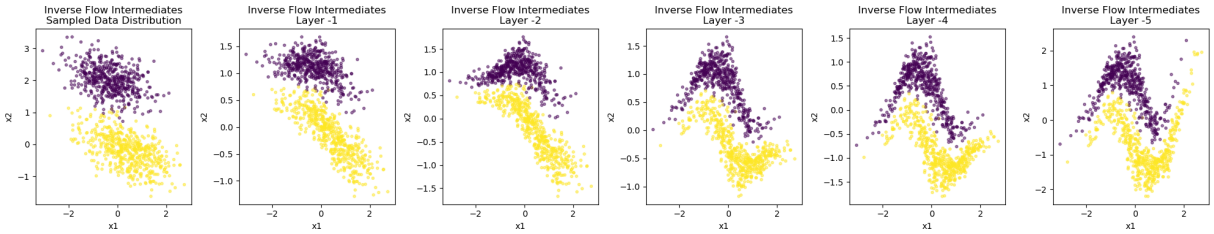
Recent advancements [3, 9, 12, 13, 16, 22, 40, 42] in IDS have focused on integrating machine learning (ML) and deep learning (DL) techniques to enhance detection capabilities. These approaches leverage large datasets to train models capable of identifying complex patterns indicative of malicious activity. Although ML and DL methods have demonstrated superior performance in detecting various types of attacks, they often rely heavily on labeled data, which is costly and time-consuming to acquire [52]. Furthermore, the inherent data imbalance in real-world network traffic, where benign activities vastly outnumber malicious ones, poses a significant challenge for these methods. This imbalance can result in biased models that underperform in detecting rare but critical threats.

Machine learning and deep learning have become integral to modern intrusion detection due to their ability to process vast amounts of data and uncover hidden patterns [3, 30]. Supervised learning techniques, such as Support Vector Machines (SVM), Decision Trees [27, 33], and various neural network architectures [4, 38, 44, 46], have been extensively applied in IDS. These methods, however, require a large amount of labeled data for training, which is often impractical to acquire in real-world scenarios. Consequently, the performance of supervised methods is heavily dependent on the quality and quantity of labeled data available.

Deep learning, with its capacity for automatic feature extraction and high-dimensional data processing [8, 51], has shown significant promise in enhancing intrusion detection accuracy. Techniques such as Convolutional Neural Networks (CNNs) [26, 44] and Recurrent Neural Networks (RNNs) [23, 49] have been employed to analyze network traffic and identify malicious patterns. However, these methods also suffer from the data imbalance issue, where the overrepresentation of benign traffic results in models that are less effective at detecting rare attacks. Despite various attempts to address this issue, including data augmentation, resampling, and cost-sensitive learning, a robust solution for handling data imbalance in semi-supervised settings remains elusive.



(a) Forward flow.



(b) Inverse flow.

Figure 2: Intermediate steps in Normalizing flows.

2.2 Normalizing Flows

Normalizing flows have recently gained attention as a promising approach for modeling complex data distributions [31]. These models consist of a series of invertible transformations that map a simple distribution (e.g., Gaussian) to a more complex one, enabling exact likelihood estimation and efficient sampling. The flexibility and scalability of normalizing flows make them particularly well-suited for high-dimensional data, such as network traffic, where capturing intricate patterns is crucial for effective intrusion detection [1, 19].

Despite their potential, the application of normalizing flows in intrusion detection is still in its early stages. Current research has explored their use in unsupervised and semi-supervised settings, demonstrating their ability to model the distribution of benign traffic and identify anomalies as deviations from this distribution. However, existing studies have not fully addressed the challenges posed by data imbalance in intrusion detection. There remains a need for more comprehensive approaches that integrate normalizing flows with strategies specifically designed to handle imbalanced data distributions.

Normalizing flows differ from traditional machine learning and deep learning methods in several key aspects. Unlike standard neural networks, which often operate as black boxes, normalizing flows provide a tractable way to estimate the likelihood of data points, facilitating a better understanding and interpretation of the learned models. This characteristic is particularly advantageous in semi-supervised learning, where normalizing flows can leverage both labeled and unlabeled data more effectively. By modeling the underlying data distribution more accurately, normalizing flows can enhance the detection of rare and novel attacks, thereby addressing the limitations of existing methods in handling data imbalance.

Given these advantages, our proposed method integrates normalizing flows into a semi-supervised learning framework for intrusion detection. This approach leverages the strengths of normalizing flows in modeling complex data distributions and mitigating data imbalance. By combining self-supervised learning and adversarial training strategies, we enhance the model's adaptability to imbalanced unlabeled data, improving overall detection performance. This work contributes to the field by offering a novel solution to the longstanding challenge of data imbalance in intrusion detection, demonstrating its effectiveness through extensive experiments on real-world datasets.

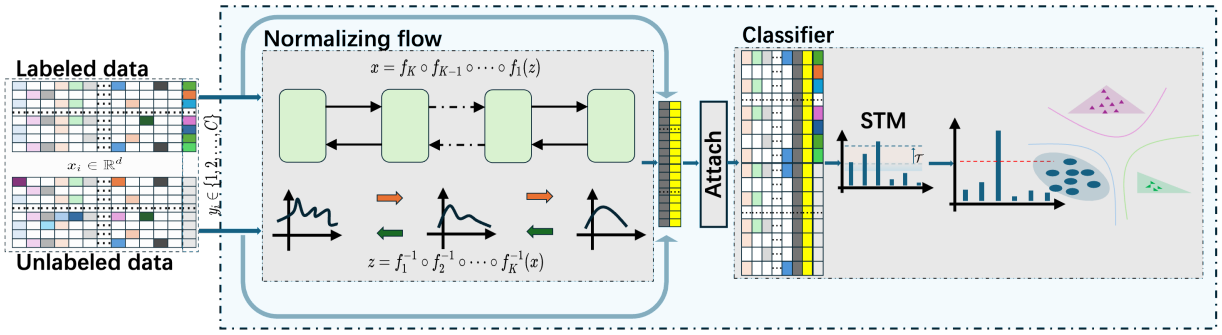


Figure 3: Normalizing flow-based semi-supervised intrusion detection. Sensitivity Threshold Management (STM) controls the inclusion of instances in the loss computation based on their uncertainty measures.

3 Methodology

Our proposed method utilizes normalizing flows to address the challenge of data imbalance in network intrusion detection. This approach is specifically designed to handle both labeled and unlabeled data, thereby enhancing detection performance in scenarios where data distributions are imbalanced. By integrating normalizing flows with semi-supervised learning techniques, our method aims to improve the robustness and scalability of intrusion detection systems. The key components of the proposed method are outlined below.

3.1 Problem Formulation

In this section, we present the mathematical formulation of semi-supervised learning for intrusion detection, addressing the imbalanced distribution of both labeled and unlabeled data. We then introduce the concept of normalizing flows and detail their application in learning from imbalanced, unlabeled data.

Semi-Supervised Learning in Intrusion Detection: Let $\mathcal{D} = \mathcal{D}_L \cup \mathcal{D}_U$ represent the dataset, where $\mathcal{D}_L = \{(x_i, y_i)\}_{i=1}^{N_L}$ is the labeled subset with N_L labeled instances and $\mathcal{D}_U = \{x_i\}_{i=N_L+1}^{N}$ is the unlabeled subset with $N_U = N - N_L$ unlabeled instances. Here, $x_i \in \mathbb{R}^d$ represents a feature vector and $y_i \in \{1, 2, \ldots, C\}$ is the corresponding class label, where C is the number of classes also means different intrusions.

Data Imbalance in Intrusion Detection: Let N_{max} and N_{min} denote the number of instances in the most and least frequent classes in $\mathcal{D}_L$, respectively. The imbalance ratio γ_L for the labeled data is defined as: $\gamma_L = N_{max}/N_{min}$. Similarly, the imbalance ratio γ_U for the unlabeled data $\mathcal{D}_U$ can be defined if the class distribution is inferred or known. For multi-classification, we use exponential sampling to sample data categories. The sampling results are shown in Figure 1, the "Intrusion Types" on the horizontal axis represents different types of network intrusions like DoS, Probe, R2L, U2R, and normal traffic. The vertical axis represents the number of samples. The exponential sampling method can effectively balance the distribution of different types of network intrusions.

Normalizing Flows: Normalizing flows provide a method to construct complex probability distributions by transforming a simple base distribution through a series of invertible mappings shown in Figure 2. Formally, let $z \sim p_Z(z)$ be a sample from a base distribution p_Z. A normalizing flow f is a bijective function that transforms z into x such that $x = f(z)$ and $z = f^{-1}(x)$ also expressed as $x = f_K \circ f_{K-1} \circ \cdots \circ f_1(z_0)$ and $z_0 = f_1^{-1} \circ f_2^{-1} \circ \cdots \circ f_K^{-1}(x)$ with a flow composed of K transformations. The probability density function $p_X(x)$ of x is given by:

$$p_X(x) = p_Z(z) \left| \det \frac{\partial f^{-1}(x)}{\partial x} \right| = p_Z(f^{-1}(x)) \cdot \left| \det \frac{\partial f^{-1}(x)}{\partial x} \right|. \tag{1}$$

From the simple distribution z_0 to the final observed variable x, the intermediate latent variable is z_k:

$$z_0 \to z_1 \to \cdots \to z_k \to x. \tag{2}$$

The transformation of each (k) layer is defined as:

$$z_k = f_k(z_{k-1}) \leftrightarrows z_{k-1} = f_k^{-1}(z_k), \quad k = 1, 2, \ldots, K. \tag{3}$$

For a flow composed of K transformations, the log-likelihood of the data is given by:

$$\log p_X(x) = \log p_Z(z_0) + \sum_{k=1}^{K} \log \left| \det \frac{\partial f_k(z_{k-1})}{\partial z_{k-1}} \right|, \tag{4}$$

Where $\left|\det \frac{\partial f_k(z_{k-1})}{\partial z_{k-1}}\right|$ is the determinant of the Jacobian of the inverse transformation, $p_Z(z)$ is the base distribution. Using Bayes' theorem, the posterior distribution of z given x is:

$$p_Z(z|x) = \frac{p_X(x|z)p_Z(z)}{p_X(x)} \tag{5}$$

The objective is to model the complex distribution of network traffic data using normalizing flows. The normalizing flow is trained to maximize the likelihood of the data under the model, which involves optimizing the parameters of the bijective transformations to fit the data distribution.

Applying Normalizing Flows to Intrusion Detection: In a semi-supervised intrusion detection setting, normalizing flows can be used to learn the distribution of both labeled and unlabeled data. The process involves the following steps:

Base distribution selection and flow construction: Choose a simple distribution $p_Z(z)$, typically a multivariate Gaussian and then define a sequence of invertible transformations $f = f_K \circ f_{K-1} \circ \cdots \circ f_1$, where each f_k is a simple, bijective mapping.

Use the labeled data $\mathcal{D}_L$ to train the normalizing flow by maximizing the log-likelihood, and use the unlabeled data $\mathcal{D}_U$ to further refine the flow model. This can be done by semi-supervised learning strategies such as pseudo-labeling or self-training, where the model's predictions on $\mathcal{D}_U$ are used to iteratively improve the model.

To address data imbalance, incorporate techniques such as class-balanced sampling[10, 34], weighted loss functions[15, 39], or adversarial training[6, 36] to ensure the model does not become biased towards the majority class. For instance, a weighted log-likelihood can be used where classes are weighted inversely proportional to their frequencies.

By integrating these steps, the normalizing flow can effectively model the underlying distribution of both labeled and unlabeled data, enhancing the detection of rare and novel intrusions in the presence of data imbalance.

3.2 Proposed Method

Our proposed method leverages normalizing flows within a semi-supervised learning framework to address the data imbalance issue in intrusion detection. By accurately modeling the complex distribution of network traffic data and incorporating both labeled and unlabeled data, our approach improves the detection performance and robustness of intrusion detection systems. The framework depicted in Figure 3 outlines the key component of the proposed method. These modules address the challenges posed by evolving data distributions, the need for discriminative embedding learning, and the management of uncertainty in unlabeled data.

3.2.1 Pseudo Labeling

Pseudo labeling plays a crucial role in our semi-supervised learning framework by effectively utilizing unlabeled data to enhance the intrusion detection capability of the model. For each unlabeled sample, the normalizing flow model maps the input to a latent space, where the class probabilities are computed. The predicted label, or pseudo label, is assigned based on the highest probability:

$$\tilde{y} = \arg\max_c p(y = c \mid f(x_u)), \tag{6}$$

where $f(x_u)$ is the latent representation of the unlabeled sample x_u, and $p(y = c \mid f(x_u))$ is the class probability derived from the classifier in the latent space. Here c is the class index.

To ensure the reliability of pseudo labels, we employ confidence-based filtering. Only samples with high confidence predictions, defined as $\max(p(y \mid f(x_u))) \geq \tau$, are retained for training, where τ is a pre-defined confidence threshold. This filtering mechanism prevents the model from being misled by low-confidence or noisy pseudo labels. The pseudo-labeled samples contribute to the overall training process via a pseudo-label loss, defined as:

$$\mathcal{L}_{\text{pseudo}} = -\frac{1}{|\mathcal{D}_{pseudo}|} \sum_{x_u \in \mathcal{D}_{pseudo}} \sum_{c=1}^{C} \mathbb{I}[\tilde{y} = c] \log p(y = c \mid f(x_u)), \tag{7}$$

where $\mathcal{D}_{pseudo}$ is the set of confidently pseudo-labeled samples, C is the number of classes, and $\mathbb{I}$ is the indicator function, c is the class index.

3.2.2 Sensitivity Threshold Management

Despite the effectiveness of pseudo labeling, we observed during experiments that using a fixed high confidence threshold (τ) limits the inclusion of a sufficient number of unlabeled samples. This restriction hampers the model's ability to learn robust data representations, particularly in scenarios with a highly imbalanced data distribution or scarce labeled samples. To address this issue, we introduce the Sensitivity Threshold Management (STM) module, which dynamically adjusts the confidence threshold during training.

The module adaptively modulates the threshold τ based on the pseudo label distribution and the training progress. Initially, a lower threshold is used to incorporate more unlabeled samples, gradually increasing to enforce stricter confidence requirements as the model becomes more stable. The threshold at epoch t is computed as:

$$\tau_t = \tau_{min} + (\tau_{max} - \tau_{min}) \cdot \frac{t}{T}, \tag{8}$$

where τ_{min} and τ_{max} are the minimum and maximum thresholds, t is the current epoch, and T is the total number of training epochs.

By lowering the threshold in the early stages of training, STM allows the model to leverage a broader set of unlabeled samples, facilitating better exploration of the data distribution. As the training progresses, the increasing threshold ensures that only high-confidence samples are used, improving the quality of pseudo labels and preventing noisy samples from impacting the model.

With STM, the pseudo-label loss is modified to account for the dynamic threshold:

$$\mathcal{L}_{pseudo} = -\frac{1}{|\mathcal{D}_{pseudo,t}|} \sum_{x_u \in \mathcal{D}_{pseudo,t}} \sum_{c=1}^{C} \mathbb{I}[\tilde{y} = c] \log p(y = c \mid f(x_u)), \tag{9}$$

where $\mathcal{D}_{pseudo,t}$ is the set of pseudo-labeled samples at epoch t that satisfy the dynamic threshold τ_t.

The STM module improves the adaptability of the model to different data distributions and imbalance levels in intrusion detection. By dynamically controlling the sensitivity of pseudo labeling, the proposed framework achieves a better trade-off between data utilization and label quality, ultimately enhancing detection performance and robustness.

Pseudo labeling is integrated into the proposed framework alongside supervised learning on labeled data. The total loss function is defined as:

$$\mathcal{L} = \mathcal{L}_{\text{supervised}} + \alpha \mathcal{L}_{\text{pseudo}} + \beta \mathcal{L}_{\text{flow}}, \tag{10}$$

where $\mathcal{L}_{supervised}$ is the supervised loss on labeled data, and α is a weighting factor balancing the contribution of pseudo labels, A regularization term $\mathcal{L}_{\text{flow}}$ is introduced to ensure that the distribution of the latent representation z is close to the prior distribution of the flow.

In the context of intrusion detection, pseudo labeling enables the model to incorporate vast amounts of unlabeled network traffic data, thereby improving its generalization and adaptability to unseen attacks. Moreover, by leveraging the expressive power of normalizing flows to accurately capture

the underlying distribution of network traffic, pseudo labeling helps the framework maintain robust performance even in the presence of data imbalance and evolving attack patterns. Pseudo code see Table 1.

Table 1: Optimized Pseudo Code for Semi-Supervised Training with Normalizing Flow and STM

Step	Description
1	Initialize Normalizing Flow model f, optimizer, and scheduler.
2	For each epoch $t \in \{1, \ldots, T\}$:
3	Compute dynamic threshold $\tau_t = \tau_{\min} + (\tau_{\max} - \tau_{\min}) \cdot \frac{t}{T}$.
4	For each batch $(x_l, y_l) \in \mathcal{D}_{\text{labeled}}, (x_u, \_) \in \mathcal{D}_{\text{unlabeled}}$:
5	*Supervised Loss:*
6	Compute $\mathcal{L}_{\text{supervised}}$ using labeled data (x_l, y_l).
7	*Pseudo Labeling with STM:*
8	Compute pseudo labels $\tilde{y}$ and confidence scores for x_u.
9	Select samples x_u with high confidence $> \tau_t$ and compute $\mathcal{L}_{\text{pseudo}}$.
10	*Latent Regularization:*
11	Compute $\mathcal{L}_{\text{flow}}$ using latent representations.
12	For labeled and pseudo-labeled data, ensure latent z aligns with prior $p(z)$:
13	$\mathcal{L}_{\text{flow}} = -\frac{1}{N} \sum_{i=1}^{N} [\log p(z_i) + \log\lvert\det J_i\rvert]$,
14	where $p(z)$ is a Gaussian Mixture Model prior, $\lvert\det J_i\rvert$ is Jacobian determinant, N is the number of the training data.
15	*Total Loss and Gradient Update:*
16	Combine all losses:
17	$\mathcal{L} = \mathcal{L}_{\text{supervised}} + \alpha \mathcal{L}_{\text{pseudo}} + \beta \mathcal{L}_{\text{flow}}$.
18	Perform back propagation and update model parameters.
19	Log metrics: loss, accuracy, precision, recall, and F1-score on validation data.
20	Return the trained Normalizing Flow model $f(\cdot)$.

4 Experiments

In this section, we detail the experimental setup, present the results, and conduct ablation studies to demonstrate the effectiveness and robustness of our proposed method under data imbalance conditions.

Datasets: We conduct our experiments on several widely used intrusion detection datasets. NSL-KDD[14]: an improved version of KDD Cup 99[41], addressing some of its inherent issues. CICIDS2018[37]: a comprehensive dataset containing realistic network traffic with various attack types. UNSW-NB15[29]: a more recent dataset with a diverse set of attack types and normal traffic. **Evaluation Metrics:** Given the focus on imbalanced data, we employ the following evaluation metrics. *Accuracy*: The proportion of correctly identified instances over the total instances. *Precision*: The proportion of correctly identified attacks out of all instances predicted as attacks. *Recall*: The proportion of actual attacks correctly identified by the model. $F_{1-score}$: The harmonic mean of precision and recall, providing a balanced measure of performance. **Baselines:** We compare our method against several state-of-the-art baselines for intrusion detection. Support Vector Machines (SVM), Random Forest (RF), and Deep Neural Networks (DNN), Variational AutoEncoders (VAE), Generative Adversarial Networks (GAN). **Implementation Details:** Preprocessing: we normalize all features to have zero mean and unit variance. Categorical features are encoded using one-hot encoding. Training: we use a learning rate of $1e-3$, batch size of 128, and the Adam optimizer and scheduler with a CosineAnnealingLR strategy for training our models. Hyperparameters: The weighting factors for the loss components α, β are selected through cross-validation. For different datasets and settings, we adjust the hyperparameters to achieve optimal performance and robustness. The K for the normalizing flow is set to 3 for dataset NSL-KDD, and the latent dimension is 64, the n-components is 3, noted as 3-64-3. For dataset CICIDS2018 is 5-128-3, for dataset UNSW-NB15

is 5-64-5. The confidence threshold τ is initialized at 0.5 and dynamically adjusted using the STM module. The total number of training epochs is set to 500.

Table 2: Results on NSL-KDD with different unlabeled data distributions.

Method	NSL-KDD											
	N_l=500/N_u=500, i_l=100/i_u=100				N_l=500/N_u=500, i_l=100/i_u=1				N_l=500/N_u=500, i_l=100/i_u=100/R			
	Acc.	*Rec.*	*Pre.*	F_1	*Acc.*	*Rec.*	*Pre.*	F_1	*Acc.*	*Rec.*	*Pre.*	F_1
SVM	0.76	0.68	0.76	0.72	0.71	0.61	0.74	0.71	0.69	0.63	0.70	0.68
RF	0.78	0.69	0.74	0.73	0.73	0.70	0.76	0.73	0.72	0.69	0.75	0.72
DNN	0.79	0.72	0.81	0.77	0.75	0.68	0.77	0.72	0.73	0.67	0.79	0.73
VAE	0.74	0.62	0.71	0.69	0.69	0.60	0.68	0.65	0.67	0.61	0.66	0.64
GAN	0.75	0.71	0.73	0.72	0.70	0.63	0.71	0.68	0.68	0.64	0.70	0.67
Ours	0.81	0.75	0.79	0.78	0.76	0.70	0.78	0.75	0.75	0.69	0.74	0.73

* N_l: number of labeled instances, N_u: number of unlabeled instances, i_l: imbalance ratio of labeled instances, i_u: imbalance ratio of unlabeled instances, R: reverse distribution, *Acc.*: *accuracy*, *Rec.*: *recall*, *Pre.*: *precision*, F_1: $F_{1-score}$.

Table 3: Results on UNSW-NB15 with same distribution.

Method	UNSW-NB15											
	N_l=500/N_u=1000, i_l=100/i_u=100				N_l=500/N_u=1000, i_l=200/i_u=200				N_l=500/N_u=1000, i_l=500/i_u=500			
	Acc.	*Rec.*	*Pre.*	F_1	*Acc.*	*Rec.*	*Pre.*	F_1	*Acc.*	*Rec.*	*Pre.*	F_1
SVM	0.70	0.65	0.70	0.69	0.68	0.62	0.73	0.66	0.65	0.61	0.70	0.64
RF	0.72	0.64	0.71	0.67	0.69	0.66	0.70	0.67	0.67	0.63	0.68	0.65
DNN	0.73	0.70	0.76	0.73	0.71	0.65	0.74	0.69	0.69	0.62	0.71	0.67
VAE	0.68	0.61	0.66	0.65	0.65	0.61	0.64	0.63	0.60	0.53	0.61	0.59
GAN	0.69	0.57	0.67	0.63	0.66	0.62	0.66	0.64	0.64	0.57	0.62	0.61
Ours	0.74	0.69	0.76	0.73	0.73	0.67	0.72	0.69	0.70	0.62	0.71	0.67

Table 4: Results on CICIDS2018 with Improved Performance as Unlabeled Data Increases.

Method	CICIDS2018											
	N_l=500/N_u=1000, i_l=500/i_u=1000				N_l=500/N_u=1000, i_l=500/i_u=500				N_l=500/N_u=1000, i_l=500/i_u=100			
	Acc.	*Rec.*	*Pre.*	F_1	*Acc.*	*Rec.*	*Pre.*	F_1	*Acc.*	*Rec.*	*Pre.*	F_1
SVM	0.74	0.67	0.73	0.72	0.76	0.72	0.75	0.74	0.78	0.73	0.77	0.76
RF	0.75	0.70	0.74	0.73	0.78	0.73	0.76	0.75	0.80	0.75	0.79	0.78
DNN	0.76	0.69	0.74	0.74	0.80	0.74	0.78	0.77	0.82	0.76	0.81	0.80
VAE	0.70	0.67	0.69	0.68	0.72	0.67	0.71	0.70	0.75	0.70	0.74	0.73
GAN	0.71	0.64	0.70	0.69	0.74	0.69	0.72	0.71	0.76	0.71	0.75	0.74
Ours	0.78	0.72	0.77	0.76	0.83	0.78	0.81	0.80	0.84	0.79	0.84	0.83

4.1 Experimental Results

Table 2 shows the results on NSL-KDD dataset with different settings. The proposed method consistently outperforms the baselines across all metrics, demonstrating its effectiveness in handling imbalanced data distributions. The adaptive marginal distribution adjustment module plays a crucial role in adapting to evolving data characteristics, while the contrastive learning integration enhances the discriminative power of the model. The sensitivity threshold management module further improves the model's robustness against noisy labels and uncertainties, contributing to its overall performance. The results on UNSW-NB15 and CICIDS2018 datasets are presented in Tables 3 and 4, respectively. Our

method achieves superior performance compared to the baselines on these datasets as well, highlighting its effectiveness in handling imbalanced data distributions and enhancing intrusion detection accuracy.

4.2 Ablation Study

To evaluate the contribution of each module in our method, we conduct ablation studies by systematically removing or modifying specific modules and observing the impact on performance, the result see Table 5.

Sensitivity threshold management: Excluding this module results in increased model sensitivity to noisy labels, causing a decline in precision and overall accuracy. By dynamically adjusting the confidence threshold, the model can effectively filter out low-confidence pseudo labels, improving the quality of training data and enhancing detection performance. Without pseudo labeling: Removing the pseudo labeling module leads to a significant drop in recall and F1-score, indicating the importance of leveraging unlabeled data for intrusion detection. By incorporating pseudo labels, the model can effectively utilize unlabeled samples to enhance its detection capability, particularly in scenarios with imbalanced data distributions.

The ablation study results confirm that each module of our proposed method contributes significantly to its overall performance. The full model configuration consistently achieves the best results, underscoring the necessity of each module.

Table 5: Ablation study on NSL-KDD dataset with different module settings.

Dataset and settings	Methods	Modules		Metrics			
NSL-KDD	Full Supervised	Pseudo Label	STM	*Acc.*	*Pre.*	*Rec.*	F_1
with all labeled data	✓	-	-	0.81	0.71	0.58	0.60
N_l=500/N_u=1000 i_l=100/i_u=100	-	✓	✓	0.81	0.73	0.79	0.78
	-	✓	×	0.78	0.68	0.75	0.72
	-	×	×	0.72	0.62	0.70	0.65

For normalizing flow settings, we observed that increasing the number of flow layers and latent dimensions improves the model's capacity to capture the complex distribution of network traffic data. However, excessively large latent dimensions may lead to overfitting, while too few layers may limit the model's expressiveness. Increasing the number of Gaussian mixture model components will not improve the effect, but will cause overfitting. By tuning these hyperparameters, we achieve a balance between model complexity and generalization performance.

Table 6: Ablation study on NSL-KDD dataset with different flow settings.

Dataset and settings	Flow Settings			Metrics			
NSL-KDD	K-flow	N-Latent Dimension	N-components	*Acc.*	*Pre.*	*Rec.*	F_1
report result	3	64	3	0.81	0.73	0.79	0.78
N_l=500/N_u=1000 i_l=100/i_u=100	2	16	3	0.57	0.43	0.65	0.52
	2	32	3	0.72	0.64	0.57	0.60
	2	64	3	0.78	0.69	0.72	0.71
	3	64	5	0.74	0.61	0.70	0.66
	3	128	3	0.75	0.64	0.73	0.67
	5	64	3	0.76	0.63	0.74	0.69
	5	128	3	0.69	0.61	0.67	0.63

* K-flow: number of flow layers, N-Latent Dimension: latent dimension, N-components: number of Gaussian mixture model components.

5 Conclusion

In this paper, we propose a novel semi-supervised learning framework for intrusion detection that effectively addresses the challenge of data imbalance, a prevalent issue in intrusion detection scenarios.

Our approach utilizes normalizing flows to model the distribution of both labeled and unlabeled data, incorporating sensitivity threshold management to enhance detection performance.

Experimental results on multiple benchmark datasets demonstrate the effectiveness and robustness of our method. Our approach consistently outperforms state-of-the-art baselines across most evaluation metrics, particularly under conditions of imbalanced data. The ablation studies further validate the significance of each component, highlighting their collective contribution to the overall performance of the model.

Our method not only improves the accuracy of intrusion detection systems but also ensures robustness against the varying distributions of unlabeled data—a common issue in real-world applications. By dynamically adapting to the evolving data distribution and incorporating contrastive learning to learn discriminative embeddings, our approach provides a comprehensive solution to the challenges posed by semi-supervised learning in the context of intrusion detection.

6 Discussion

Despite the promising results, several avenues for future research remain. First, extending our framework to handle more diverse and dynamic network environments would be valuable. This includes evaluating the approach on real-time streaming data to assess its performance in live network conditions. Additionally, further exploration of more sophisticated data augmentation techniques to generate realistic variations of network traffic could enhance the robustness of the model.

Another potential direction is the integration of domain adaptation techniques to improve the model's ability to generalize across different network domains with varying characteristics. Finally, investigating the incorporation of more advanced neural architectures and optimization strategies could further enhance the performance and efficiency of the proposed method.

Our work makes a significant contribution to the field of intrusion detection by providing a robust and effective solution to the challenges posed by data imbalance and the semi-supervised learning paradigm. We believe that the proposed framework can serve as a foundation for future developments in this area, paving the way for more accurate and reliable intrusion detection systems.

Acknowledgements

We would like to express our sincere gratitude to our colleagues for their valuable feedback and support throughout this research. We also extend our appreciation to the anonymous reviewers for their insightful comments and constructive suggestions.

Author contributions

The authors contributed equally to this work.

Conflict of interest

The authors declare no conflict of interest.

References

[1] Abuadlla, Y., Kvascev, G., Gajin, S. Jovanovic, Z. (2014). Flow-based anomaly intrusion detection system using two neural network stages. *Computer Science And Information Systems*. 11, 601-622, 2014.

[2] Adadi, A. (2021). A survey on data-efficient algorithms in big data era. *Journal Of Big Data*. 8, 24, 2021.

[3] Ahmad, Z., Shahid Khan, A., Wai Shiang, C., Abdullah, J. Ahmad, F. (2021). Network intrusion detection system: A systematic study of machine learning and deep learning approaches. *Transactions On Emerging Telecommunications Technologies*. 32, e4150, 2021.

[4] Atefinia, R. Ahmadi, M. (2021). Network intrusion detection using multi-architectural modular deep neural network. *The Journal Of Supercomputing*. 77, 3571-3593, 2021.

[5] Bagui, S. Li, K. (2021). Resampling imbalanced data for network intrusion detection datasets. *Journal Of Big Data*. 8, 6, 2021.

[6] Bai, T., Luo, J., Zhao, J., Wen, B. Wang, Q. (2021). Recent advances in adversarial training for adversarial robustness. *ArXiv Preprint ArXiv:2102.01356*, 2021.

[7] Bhosale, D. Mane, V. (2015). Comparative study and analysis of network intrusion detection tools. *2015 International Conference On Applied And Theoretical Computing And Communication Technology (iCATccT)*. 312-315, 2015.

[8] Bolón-Canedo, V., Sánchez-Maroño, N. Alonso-Betanzos, A. (2016). Feature selection for high-dimensional data. *Progress In Artificial Intelligence*. 5, 65-75, 2016.

[9] Chiba, Z., Abghour, N., Moussaid, K., Rida, M. (2019). Others Intelligent approach to build a Deep Neural Network based IDS for cloud environment using combination of machine learning algorithms. *Computers & Security*. 86, 291-317, 2019.

[10] Cui, Y., Jia, M., Lin, T., Song, Y. Belongie, S. (2019). Class-balanced loss based on effective number of samples. *Proceedings Of The IEEE/CVF Conference On Computer Vision And Pattern Recognition*. 9268-9277, 2019.

[11] Das, S., Mullick, S. Zelinka, I. (2022). On supervised class-imbalanced learning: An updated perspective and some key challenges. *IEEE Transactions On Artificial Intelligence*. 3, 973-993, 2022.

[12] Zhang, D. (2017). High-speed train control system big data analysis based on the fuzzy RDF model and uncertain reasoning. *International Journal Of Computers Communications & Control*. 12, 577-591, 2017.

[13] Zhang, D., Du, C., Peng, Y., Liu, J., Mohammed, S. & Calvi, A. (2024). A multi-source dynamic temporal point process model for train delay prediction. *IEEE Transactions On Intelligent Transportation Systems*. 2024.

[14] Dhanabal, L. Shantharajah, S. (2015). A study on NSL-KDD dataset for intrusion detection system based on classification algorithms. *International Journal Of Advanced Research In Computer And Communication Engineering*. 4, 446-452, 2015.

[15] Gong, T., Lee, T., Stephenson, C., Renduchintala, V., Padhy, S., Ndirango, A., Keskin, G. Elibol, O. (2019). A comparison of loss weighting strategies for multi task learning in deep neural networks. *IEEE Access*. 7, 141627-141632, 2019.

[16] Hamdi, M., Bouhamed, H., Badreddine, F. & Alkanhel, R. (2024) Deep recurrent neural networks distributed on a Hadoop/Spark cluster for fall detection: Deep recurrent neural networks for fall detection. *Int. J. Comput. Commun. Control*. 19, (2024), https://doi.org/10.15837/ijccc.2024.3.6428

[17] Hajj, S., El Sibai, R., Bou Abdo, J., Demerjian, J., Makhoul, A. Guyeux, C. (2021). Anomaly-based intrusion detection systems: The requirements, methods, measurements, and datasets. *Transactions On Emerging Telecommunications Technologies*. 32, e4240, 2021.

[18] He, H. Garcia, E. (2009). Learning from imbalanced data. *IEEE Transactions On Knowledge And Data Engineering*. 21, 1263-1284, 2009.

[19] Idrissi, M., Alami, H., Bouayad, A. Berrada, I. (2023). NF-NIDS: Normalizing Flows for Network Intrusion Detection Systems. *2023 10th International Conference On Wireless Networks And Mobile Communications*. 1-7, 2023.

[20] Ioulianou, P., Vasilakis, V., Moscholios, I. Logothetis, M. (2018). A signature-based intrusion detection system for the internet of things. *Information And Communication Technology Form.* 2018.

[21] Jha, J. Ragha, L. (2013). Intrusion detection system using support vector machine. *International Journal Of Applied Information Systems*. 3, 25-30, 2013.

[22] Kasongo, S. Sun, Y. (2019). A deep learning method with filter based feature engineering for wireless intrusion detection system. *IEEE Access*. 7, 38597-38607, 2019.

[23] Kim, J., Kim, J., Thu, H. Kim, H. (2016). Long short term memory recurrent neural network classifier for intrusion detection. *2016 International Conference On Platform Technology And Service*. 1-5, 2016.

[24] Kumar, V. Sangwan, O. (2012). Signature based intrusion detection system using SNORT. *International Journal Of Computer Applications & Information Technology*. 1, 35-41, 2012.

[25] Li, W., Tug, S., Meng, W. Wang, Y. (2019). Designing collaborative blockchained signature-based intrusion detection in IoT environments. *Future Generation Computer Systems*. 96, 481-489, 2019.

[26] Li, Z., Qin, Z., Huang, K., Yang, X. Ye, S. (2017). Intrusion detection using convolutional neural networks for representation learning. *International Conference On Neural Information Processing*. 858-866, 2017.

[27] Liu, X., Li, K., Wang, W., Yan, Y., Sha, Y., Chen, J. & Qin, J. (2021) Improved RBF Network Intrusion Detection Model Based on Edge Computing with Multi-algorithm Fusion. *Int. J. Comput. Commun. Control*. 16, 2021. https://doi.org/10.15837/ijccc.2021.4.4232

[28] Masdari, M. Khezri, H. (2020). A survey and taxonomy of the fuzzy signature-based intrusion detection systems. *Applied Soft Computing*. 92, 106301, 2020.

[29] Moustafa, N. Slay, J. (2015). UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). *2015 Military Communications And Information Systems Conference*. 1-6, 2015.

[30] Nazih, W., Hifny, Y., Elkilani, W., Abdelkader, T. & Faheem, H. (2019) Efficient Detection of Attacks in SIP Based VoIP Networks Using Linear l1-SVM Classifier. *Int. J. Comput. Commun. Control*. 14, 518-529, 2019. https://doi.org/10.15837/ijccc.2019.4.3563

[31] Papamakarios, G., Nalisnick, E., Rezende, D., Mohamed, S. Lakshminarayanan, B. (2021). Normalizing flows for probabilistic modeling and inference. *Journal Of Machine Learning Research*. 22, 1-64, 2021.

[32] Paxson, V. (1999). Bro: a system for detecting network intruders in real-time. *Computer Networks*. 31, 2435-2463, 1999.

[33] Rai, K., Devi, M. Guleria, (2016). A. Decision tree based algorithm for intrusion detection. *International Journal Of Advanced Networking And Applications*. 7, 2828, 2016.

[34] Rani, M. Singh, G. (2022). Gagandeep Effective network intrusion detection by addressing class imbalance with deep neural networks multimedia tools and applications. *Multimedia Tools And Applications*. 81, 8499-8518, 2022.

[35] Rezende, D. Mohamed, S. (2015). Variational inference with normalizing flows. *International Conference On Machine Learning*. 1530-1538, 2015.

[36] Shafahi, A., Najibi, M., Ghiasi, M., Xu, Z., Dickerson, J., Studer, C., Davis, L., Taylor, G. Goldstein, T. (2019). Adversarial training for free!. *Advances In Neural Information Processing Systems*. 32, 2019.

[37] Sharafaldin, I., Lashkari, A., Ghorbani, A. (2018). Others Toward generating a new intrusion detection dataset and intrusion traffic characterization. *International Conference on Information Systems Security and Privacy*. 1, 108-116, 2018.

[38] Shenfield, A., Day, D. Ayesh, A. (2018). Intelligent intrusion detection systems using artificial neural networks. *ICT Express*. 4, 95-99, 2018.

[39] Sudre, C., Li, W., Vercauteren, T., Ourselin, S. Jorge Cardoso, M. (2017). Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. *Deep Learning In Medical Image Analysis And Multimodal Learning For Clinical Decision Support: Third International Workshop, DLMIA 2017, And 7th International Workshop, ML-CDS 2017, Held In Conjunction With MICCAI 2017, Québec City, QC, Canada, September 14, Proceedings 3*. 240-248, 2017.

[40] S, A. & R, V. (2023) A Deep Learning Approach for Efficient Anomaly Detection in WSNs. *Int. J. Comput. Commun. Control*. 18, (2023), https://doi.org/10.15837/ijccc.2023.1.4756

[41] Tavallaee, M., Bagheri, E., Lu, W. Ghorbani, A. (2009). A detailed analysis of the KDD CUP 99 data set. *2009 IEEE Symposium On Computational Intelligence For Security And Defense Applications*. 1-6, 2009.

[42] Thakkar, A. Lohiya, R. (2021). A review on machine learning and deep learning perspectives of IDS for IoT: recent updates, security issues, and challenges. *Archives Of Computational Methods In Engineering*. 28, 3211-3243, 2021.

[43] Van Engelen, J. Hoos, H. (2020). A survey on semi-supervised learning. *Machine Learning*. 109, 373-440, 2020.

[44] Vinayakumar, R., Soman, K. Poornachandran, P. (2017). Applying convolutional neural network for network intrusion detection. *2017 International Conference On Advances In Computing, Communications And Informatics (ICACCI)*. 1222-1228, 2017.

[45] Wang, L., Han, M., Li, X., Zhang, N. Cheng, H. (2021). Review of classification methods on unbalanced data sets. *IEEE Access*. 9, 64606-64628, 2021.

[46] Wu, P. Guo, H. (2019). LuNET: a deep neural network for network intrusion detection. *2019 IEEE Symposium Series On Computational Intelligence*. 617-624, 2019.

[47] Yang, D., Usynin, A. Hines, J. (2006). Anomaly-based intrusion detection for SCADA systems. *5th Intl. Topical Meeting On Nuclear Plant Instrumentation, Control And Human Machine Interface Technologies*. 12-16, 2006.

[48] Yang, Z., Liu, X., Li, T., Wu, D., Wang, J., Zhao, Y. Han, H. (2022). A systematic literature review of methods and datasets for anomaly-based network intrusion detection. *Computers & Security*. 116, 102675, 2022.

[49] Yin, C., Zhu, Y., Fei, J. He, X. (2017). A deep learning approach for intrusion detection using recurrent neural networks. *IEEE Access*. 5, 21954-21961, 2017.

[50] Yu, H., Sun, C., Yang, X., Zheng, S., Wang, Q. Xi, X. (2018). LW-ELM: a fast and flexible cost-sensitive learning framework for classifying imbalanced data. *IEEE Access*. 6, 28488-28500, 2018.

[51] Zebari, R., Abdulazeez, A., Zeebaree, D., Zebari, D. Saeed, J. (2020). A comprehensive review of dimensionality reduction techniques for feature selection and feature extraction. *Journal Of Applied Science And Technology Trends*. 1, 56-70, 2020.

[52] Zhang, T. Oles, F. (2000). The value of unlabeled data for classification problems. *Proceedings Of The Seventeenth International Conference On Machine Learning*. 20, 10-24, 2000.

Journal's webpage: http://univagora.ro/jour/index.php/ijccc/

This journal is a member of, and subscribes to the principles of,
the Committee on Publication Ethics (COPE).
https://publicationethics.org/members/international-journal-computers-communications-and-control

Cite this paper as:
Chaoqun, Guo.; Shunjie, Yang.; Jubao, Cheng.; Dalin, Zhang., (2025). A Normalizing Flow-Based Semi-Supervised Method for Imbalanced Network Intrusion Detection, *International Journal of Computers Communications & Control*, 20(4), 6890, 2025.
https://doi.org/10.15837/ijccc.2025.4.6890