https://jsras.rcc.edu.ly/
Vol.3 No.1 (2026), 32-52
Article history:
Received: 20 April. 2026
Accepted: 10 May. 2026
Published: 11 May. 2026

Journal of Sustainable Research in Applied Sciences

Proactive Cyber Defense through Multitask Deep Learning: Real-Time Detection and Traffic Prediction in Enterprise Networks

Salma Sadek Jetlawei[1,*], saljet2020@gmail.com, https://orcid.org/0009-0009-0573-1789,
Mohamed Ali Mohamed EL-sseid[2], Moh200512@Bilim.edu.tr, https://orcid.org/0009-0007-1307-8623,
Tunç Durmuş Medeni[3], tuncmedeni@ybu.edu.tr, https://orcid.org/0000-0002-2964-3320,
İhsan Tolga Medeni[4], tolgamedeni@ybu.edu.tr, https://orcid.org/0000-0002-0642-7908,
Mansour Essgaer[5], man.essgaer@sebhau.edu.ly, https://orcid.org/0000-0002-8447-5091,
[1]Department of Computer Engineering, Higher institute of Sciences and Technology Tajoura, Libya
[2]Department of Software Engineering, Ankara Bilim University, Türkiye
[3,4]Department of Management Information Systems, Ankara Yildirim Beyazit University, Türkiye
[5]Department of Artificial Intelligence, Faculty of Information Technology, Sebha University, Sabha, Libya
DOI: https://doi.org/10.36602/jsras.2026.3.1.6
* Corresponding author saljet2020@gmail.com

Index Terms

LSTM, CESNET-TimeSeries24 dataset, network traffic, Multitask Deep Learning.

Abstract

Modern network infrastructures face escalating challenges in detecting subtle anomalies and forecasting traffic dynamics due to the high volume, velocity, and complexity of contemporary network data. This research addresses a critical gap which is a single deep learning model simultaneously and effectively perform both anomaly detection and traffic metric prediction? To answer this, we propose a novel multitask deep LSTM-based framework that unifies classification (anomaly identification) and regression (future traffic forecasting) within a shared architecture. Leveraging the real-world CESNET-TimeSeries24 dataset, this research approach exploits temporal dependencies in network flow data to jointly learn robust representations for both tasks. The key advantage of this multitask design lies in its ability to enhance generalization by sharing learned features improving anomaly detection through contextual forecasting and refining predictions by recognizing abnormal patterns. This research model achieves 95.8% classification accuracy, 76.8% anomaly recall, and an exceptional R^2 score of 0.98 in regression, demonstrating high operational reliability. This work contributes a practical, end-to-end solution that advances intelligent network monitoring by combining security and performance analytics in one system offering network operators a proactive tool for cyber defense, capacity planning, and real-time decision-making. By validating the synergy between prediction and detection, this research sets a foundation for more adaptive, efficient, and holistic AI-driven network management.

الدفاع السيبراني الاستباقي من خلال التعلم العميق متعدد المهام: الكشف في الوقت الفعلي والتنبؤ بحركة البيانات في شبكات المؤسسات

سالمة الصادق الجطلاوي[1،*]، محمد علي محمد الصيد[2]، تونج دورميش مدنى[3]، إحسان تولقا مدني[4]، منصور الصغير[5]
[1]المعهد العالي للعلوم والتقنية تاجوراء، قسم هندسة الكمبيوتر، ليبيا، saljet2020@gmail.com، https://orcid.org/0009-0009-0573-1789
[2]جامعة انقرة بلم، قسم هندسة البرمجيات، انقرة، تركيا، Moh200512@Bilim.edu.tr، https://orcid.org/0009-0007-1307-8623
[3,4]جامعة انقرة يلدرم بايزيد، قسم نظم المعلومات الإدارية، انقرة، تركيا، (tuncmedeni@ybu.edu.tr، https://orcid.org/0000-0002-2964-3320)[3]
(tolgamedeni@ybu.edu.tr، https://orcid.org/0000-0002-0642-7908)[4]
[5] جامعة سبها، قسم الذكاء الاصطناعي تقنية المعلومات، سبها، ليبيا، man.essgaer@sebhau.edu.ly، https://orcid.org/0000-0002-8447-5091
DOI: https://doi.org/10.36602/jsras.2026.3.1.6
* المؤلف المراسل saljet2020@gmail.com

الملخص

تواجه البنى التحتية الحديثة للشبكات تحديات متزايدة في اكتشاف الحالات الشاذة الدقيقة والتنبؤ بديناميكيات حركة البيانات، وذلك بسبب الحجم الكبير وسرعة البيانات وتعقيدها. يتناول هذا البحث ثغرة حاسمة، ألا وهي: هل يوجد نموذج واحد للتعلم العميق قادر على أداء كل من اكتشاف الحالات الشاذة والتنبؤ بمقاييس حركة البيانات في آن واحد وبكفاءة عالية؟ للإجابة على هذا السؤال، نقترح إطار عمل جديدًا متعدد المهام قائم على شبكة LSTM عميقة، يوحد التصنيف (تحديد الحالات الشاذة) والانحدار (التنبؤ بحركة البيانات المستقبلية) ضمن بنية مشتركة. بالاستفادة من مجموعة بيانات CESNET-TimeSeries24 الواقعية، يستغل هذا النهج البحثي التبعيات الزمنية في بيانات تدفق الشبكة لتعلم تمثيلات قوية لكلا المهمتين معًا. تكمن الميزة الرئيسية لهذا التصميم متعدد المهام في قدرته على تعزيز التعميم من خلال مشاركة الميزات المتعلمة، مما يحسن اكتشاف الحالات الشاذة عبر التنبؤ السياقي، ويحسن التنبؤات من خلال التعرف على الأنماط غير الطبيعية. يحقق هذا النموذج البحثي دقة تصنيف تصل إلى 95.8%، ونسبة استدعاء للحالات الشاذة تبلغ 76.8%، ومعامل تحديد (R^2) استثنائي قدره 0.98 في الانحدار، مما يدل على موثوقية تشغيلية عالية. يقدم هذا العمل حلاً عملياً شاملاً يُعزز مراقبة الشبكات الذكية من خلال دمج تحليلات الأمن والأداء في نظام واحد، مما يوفر لمشغلي الشبكات أداة استباقية للدفاع السيبراني، وتخطيط السعة، واتخاذ القرارات في الوقت الفعلي. ومن خلال التحقق من التآزر بين التنبؤ والكشف، يضع هذا البحث أساساً لإدارة شبكات أكثر تكيفاً وكفاءة وشمولية مدعومة بالذكاء الاصطناعي.

الكلمات المفتاحية

LSTM، مجموعة بيانات -CESNET TimeSeries24، حركة مرور الشبكة، التعلم العميق متعدد المهام.

1. INTRODUCTION

The digital infrastructure underpinning this research interconnected world is perpetually subjected to an unprecedented deluge of data [1]. Within this torrent, network traffic stands as a critical conduit, reflecting the pulse of digital interactions, service delivery, and potential vulnerabilities [2]. The ability to accurately analyze, predict, and detect anomalies within this traffic is paramount for maintaining robust network performance, ensuring cybersecurity, and optimizing resource allocation [3]. As networks evolve in scale and heterogeneity, the sheer volume and velocity of data render conventional, rule-based or shallow statistical methods increasingly inadequate for real-time, nuanced threat identification and operational insights. This research endeavors to develop and rigorously evaluate a deep learning framework, specifically employing Long Short-Term Memory (LSTM) networks, for multitask learning in the domain of network traffic analysis. The primary aim is to enhance the precision and robustness of both anomaly detection and time series prediction within complex network environments, thereby contributing to more proactive and intelligent network management. The purpose of this study is to design, implement, and evaluate a deep LSTM-based multitask learning model that can effectively analyze network traffic time series data for both anomaly detection (classification) and traffic metric prediction (regression), thereby offering a comprehensive solution for intelligent network monitoring. The significance of this research lies in its potential to provide a more sophisticated and adaptive mechanism for safeguarding critical network infrastructure and optimizing its performance. By harnessing the power of deep learning, particularly LSTMs, to model the temporal dynamics inherent in network traffic, this study contributes to: Enabling earlier and more accurate detection of novel and sophisticated cyber threats, including Distributed Denial of Service (DDoS) attacks, intrusions, as well as malware propagation [1], [2]. Facilitating proactive resource management, capacity planning, and quality of service (QoS) assurance through precise traffic forecasting. Demonstrating the practical applicability and efficacy of multitask deep learning architectures for complex real-world IT operational challenges.

1.2 Problem Statement

Modern network traffic exhibits characteristics of high dimensionality, non-stationarity, and intricate temporal dependencies, making the identification of subtle anomalies and the accurate forecasting of future states exceptionally challenging [4], [5], [6]. Existing solutions often struggle with: (1) adapting to evolving attack vectors and legitimate traffic patterns; (2) processing vast quantities of data in near real-time; and (3) simultaneously addressing multiple analytical objectives (e.g., classification of malicious activity and prediction of bandwidth utilization) within a unified framework. These limitations necessitate advanced analytical paradigms capable of learning complex representations directly from raw traffic data.

1.3 Research Objectives

The specific objectives guiding this research are:
To preprocess and transform raw network flow data into a suitable time series format for deep learning analysis.
To design and implement a deep LSTM-based neural network architecture capable of performing multitask learning, simultaneously addressing network anomaly classification and traffic metric prediction.
To train and optimize the proposed deep LSTM model using a comprehensive network traffic dataset, specifically CESNET-TimeSeries24.
To rigorously evaluate the performance of the multitask learning model using a suite of appropriate metrics for both classification, for instance, accuracy, precision, recall, F1-score) and regression, for instance, Mean Squared Error, R-squared.
To analyze the model's ability to identify anomalies and accurately predict future network traffic patterns, demonstrating its practical utility for network security and management.

1.4 Research Importance

The importance of this research extends across several critical dimensions. From a security perspective, the ability to rapidly and accurately identify anomalies in network traffic is a cornerstone of effective cyber defense. Traditional signature-based systems are often reactive, whereas a predictive and anomaly-detecting model can offer a proactive layer of defense. Operationally, precise traffic prediction enables network administrators to anticipate bandwidth demands, prevent congestion, and optimize network resources, leading to improved user experience and reduced operational costs. Furthermore, this research contributes to the broader academic discourse on applying advanced artificial intelligence techniques to solve real-world problems in information technology, particularly in the challenging domain of time series analysis within dynamic systems.

1.5 The research questions

1. How effectively can a deep LSTM-based model learn and represent the complex temporal dependencies inherent in network traffic time series data?
2. Can a multitask learning framework, utilizing deep LSTMs, simultaneously achieve high accuracy in both network anomaly detection and traffic metric prediction?
3. What is the performance of the proposed deep LSTM model in classifying anomalous network behavior when evaluated on the CESNET-TimeSeries24 dataset?
4. To what extent can the deep LSTM model accurately predict future network traffic metrics, such as the number of flows, packets, and bytes, based on historical time series data?

5. What are the key advantages and limitations of employing a deep LSTM-based multitask learning approach for real-time network traffic analysis and detection compared to traditional methods?

TABLE I. THE DATASET DESCRIPTION CESNET-TIMESERIES24

Attribute	Description
Source	Provided by CESNET (Czech national research and education network)
Domain	Network traffic analysis
Format	Time series data
Time Range	2024–2028 (5 years of recorded network traffic data)
Type	Real-world network flow records

TABLE II. THE DATASET CHARACTERISTIC AND DETAILS

Characteristic	Details
Real-World Origin	Composed of actual network flow records reflecting real-life network environments.
Temporal Structure	Data is structured as time series with observations recorded at regular intervals.
High Dimensionality	Contains 34 features per time step capturing various aspects of network behavior.
Rich Feature Set	Includes metrics like number of flows, packets, bytes, protocol ratios, direction ratios, etc.
Dynamic Nature	Exhibits significant fluctuations, trends, and spikes over time due to evolving traffic patterns.
Non-Stationarity	Statistical properties (mean, variance) change over time, requiring preprocessing/transformation.
Imbalanced Classes	Anomaly class is less frequent than normal traffic, posing challenges for detection accuracy.
Noise and Variability	Contains both legitimate surges and potential anomalies, requiring robust filtering techniques.
Multicollinearity	Some features are highly correlated (e.g., n_packets and n_bytes), suggesting redundancy.
Long-Term Dependencies	Requires modeling of temporal dependencies over extended periods for anomaly detection.
Characteristic	Details

TABLE III. THE DATASET STATISTICAL PROPERTIES

Metric	Observations
Skewness	Many features (like n_flows"
Correlation	Strong positive correlations between packet and byte counts (n_packets vs n_bytes: 0.99).
Outliers	Spikes in traffic metrics suggest presence of anomalies or unusual events.
Distribution Shape	Bell-shaped, multimodal, or skewed depending on the feature.

1.6 Literature Review

Recurrent Neural Networks (RNNs), particularly LSTMs and Gated Recurrent Units (GRUs), have emerged as prominent architectures due to their inherent ability to model sequential data. For instance, studies have explored LSTMs for intrusion detection systems (IDS). Rodriguez et al., (2021) [1] reviewed deep learning approaches for network intrusion detection, highlighting the potential of LSTMs for capturing temporal features of attack patterns. Similarly, a work by Vinayakumar et al., (2019) [2] showcased the effectiveness of deep learning models, including LSTMs, on large network datasets for anomaly detection. More recently, research has shifted towards optimizing these models and exploring multitask learning. Biazar et al., (2025) [3] proposed a hybrid deep learning model combining CNNs and LSTMs for network anomaly detection, demonstrating improved performance over standalone models. The concept of multitask learning, where a single model learns to perform multiple related tasks simultaneously, has gained traction. This paradigm is particularly appealing for network traffic analysis, where anomaly detection and traffic prediction are inherently linked. For example, Ben Dalla et al., (2025) [4] investigated multitask learning for network security, albeit not exclusively with LSTMs, emphasizing the benefits of shared representations. In the realm of time series prediction, LSTMs have consistently proven effective. Ogundokun et al., [5] on deep learning for time series forecasting underscored the superior performance of LSTMs in various domains, including financial markets and energy consumption, which translates well to network traffic. Testik et al., [6] presented an LSTM-based framework for short-term network traffic prediction, confirming its accuracy and robustness. The literature consistently points to the strength of LSTMs in handling the temporal dependencies and long-range correlations characteristic of network flow data. Several research efforts have explored various machine learning and deep learning techniques for network traffic analysis. Early works often relied on traditional machine learning algorithms such as Support Vector Machines (SVMs), Decision Trees, and K-Nearest Neighbors (KNN) for classification tasks, particularly for intrusion detection [7]. While effective for specific,

well-defined attack signatures, these methods often struggle with zero-day attacks and the dynamic nature of network traffic. With the advent of big data and increased computational power, deep learning models began to supplant traditional approaches. Convolutional Neural Networks (CNNs) have been applied for feature extraction from raw packet data [8], and Autoencoders have been utilized for unsupervised anomaly detection by learning normal traffic patterns [9]. However, for time-series data like network flows, which inherently possess temporal dependencies, RNNs, and specifically LSTMs, offer a more natural fit. This research work builds upon these foundations by explicitly integrating a multitask learning paradigm with deep LSTMs. While some studies have explored LSTMs for either anomaly detection or traffic prediction independently, the simultaneous optimization of both tasks within a single, coherent model, leveraging shared representations, remains an area with significant potential. This approach aims to capitalize on the synergistic relationship between identifying abnormal patterns and forecasting future states, leading to a more holistic and robust network intelligence system.

2. SYSTEM ARCHITECTURE

The proposed system architecture for Deep LSTM-Based Multitask Learning for Network Traffic Analysis Detection is designed to process raw network flow data, extract meaningful temporal features, and simultaneously perform anomaly classification and traffic metric prediction [10]. The architecture follows a sequential pipeline, from data ingestion to model evaluation, as depicted in Figure 2.1. Artificial Intelligence (AI) encompasses the broader field of creating intelligent machines capable of performing tasks that typically require human intelligence. Machine Learning (ML), a subset of AI, focuses on enabling systems to learn from data without explicit programming [11]. Deep Learning (DL), a specialized branch of ML, utilizes artificial neural networks with multiple layers (hence "deep") to learn complex patterns and representations directly from raw data. Unlike traditional ML, deep learning excels at automatically extracting hierarchical features, making it particularly effective for high-dimensional and complex data types such as images, natural language, and time series. Recurrent Neural Networks (RNNs) are a class of deep neural networks specifically designed to process sequential data [12]. They maintain an internal state (memory) that allows them to remember information from previous inputs in a sequence, making them suitable for tasks like speech recognition and time series forecasting [13]. Long Short-Term Memory (LSTM) networks are a special type of RNN engineered to overcome the vanishing gradient problem inherent in vanilla RNNs, enabling them to learn long-term dependencies [13], [14], [24], [25], [26]. LSTMs achieve this through a sophisticated internal structure comprising a cell state and three gates (input, forget, and output gates) that regulate the flow of information, allowing the network to selectively remember or forget information over extended sequences [14], [27], [28], [29], [30]. This capability is crucial for analyzing network traffic, where anomalies or trends might manifest over prolonged periods. The system architecture is structured around a clear data flow and processing pipeline:

The initial phase involves ingesting the raw network traffic data, specifically the CESNET-TimeSeries24 dataset. This raw data typically consists of various network flow attributes (e.g., number of flows, packets, bytes, duration, protocol types, direction ratios). Data loading mechanisms are implemented to handle the volume and format of this time series data efficiently. Once loaded, the raw data undergoes a series of preprocessing steps. Scaling numerical features to a common range (e.g., 0-1 or z-score normalization) to prevent features with larger values from dominating the learning process.

Min-Max Normalization is as below:

$$X_{\text{normalized}} = \frac{X - X_{\text{min}}}{X_{\text{max}} - X_{\text{min}}}$$

Where:

X : Original feature value

X_{min} : Minimum value of the feature across all samples

X_{max} : Maximum value of the feature across all samples

Row = [25163, 158710, 286170, 30986522, 6610, 66.77, 306.14, 54839, 553.93, 2903.68, 42665, 430.96, 2238.82, 0.58, 0.56, 0.45, 0.44, 5.22, 140.1125164, 164813, 308310, 32032555, 6520, 65.86, 310.81, 54302, 548.51, 2894.46, 40949, 413.63, 2137.91, 0.58, 0.56, 0.44, 0.44].

TABLE IV. IDENTIFY EACH FEATURE COLUMN EACH COLUMN REPRESENTS ONE FEATURE. FOR EXAMPLE AS BELOW:

Column Index	Feature Name
0	n_flows
1	n_packets
2	n_bytes
...	...
33	some_ratio_metric

Take the first column (index 0), which has value 25163. Suppose after computing min and max across all rows in that column as below:

$$X_{\text{min}} = 0$$
$$X_{\text{max}} = 172689$$

Apply normalization:

$$X_{\text{normalized}} = \frac{25163 - 0}{172689 - 0} = 0.1457$$

Repeat for all 34 features. Normalization method used is a Min-Max Scaling ([0, 1] range). Time Series Transformation is structuring the data into sequences suitable for LSTM input, often involving creating lagged features or sliding windows to capture temporal context.

Mathematical Representation via given raw data matrix $D \in \mathbb{R}^{T \times F}$,

where as below:

T : Number of time points

F : Number of features (e.g., 34)

This research study construct input-output pairs:

$$X_i = [d_i, d_{i+1}, \dots, d_{i+L-1}], Y_i = d_{i+L} \quad [14].$$

Where:

L : Sequence length (window size)

$X_i \in \mathbb{R}^{L \times F}$

$Y_i \in \mathbb{R}^F$

Total number of samples: $N = T - L - 1$

TABLE V. TRANSFORMATION PROCESS

Stage	Description
Input	Flat 2D CSV file (29.csv) with 34 features over time
Preprocessing	Normalize/standardize features using Min-Max or Z-score
Sequence Creation	Sliding window of fixed length (e.g., 10) to form 3D sequences
Output Shape	(N_samples, sequence_length, n_features)
Model Ready	Can be fed directly into LSTM-based multitask model

Feature Selection/Reduction is potentially applying techniques like Principal Component Analysis (PCA) to reduce dimensionality and noise, as suggested by the PCA analysis in Figure 6. This research has applied Principal Component Analysis (PCA) for feature selection and dimensionality reduction, the mathematical transformation that projects the original high-dimensional feature space into a lower-dimensional subspace while preserving as much variance as possible. The network traffic analysis within 34 features with potential multicollinearity [15], noise, or redundancy. PCA helps reduce these to fewer uncorrelated components that still capture.

Input data matrix $X \in \mathbb{R}^{n \times p}$

n : Number of time points (rows)

$p = 34$: Number of features per time point

All features are normalized/standardized (as previously done)

Standardize the Data

Standardization ensures each feature has zero mean and unit variance:

$$z_{ij} = \frac{x_{ij} - \mu_j}{\sigma_j} \quad [15], [16], [17].$$

Where:

x_{ij} : Original value at row i, column j

μ_j : Mean of feature j

σ_j : Standard deviation of feature j

The covariance matrix $C \in \mathbb{R}^{p \times p}$ captures how each pair of features varies together:

$$C = \frac{1}{n-1} X^T X \quad [17].$$

X: Standardized data matrix

C: Symmetric covariance matrix

This research compute eigenvalues $\lambda_1, \lambda_2, \dots, \lambda_p$ and corresponding eigenvectors $v_1, v_2, \dots, v_p$ from the covariance matrix C. Each eigenvector represents a principal component, and its corresponding eigenvalue indicates [18], [19], [20], [39] how much variance is explained by that component.

$$V = [v_1, v_2, \dots, v_p] \in \mathbb{R}^{p \times p} \quad [18].$$

Matrix of eigenvectors (sorted descending by eigenvalues)

Select Top k Principal Components

Choose the top k eigenvectors that explain a desired percentage of total variance (e.g., 95%).

Compute cumulative explained variance ratio:

$$\text{Explained Variance Ratio} = \left[\frac{\lambda_1}{\sum_{i-1}^{p} \lambda_i}, \frac{\lambda_2}{\sum_{i-1}^{p} \lambda_i}, \dots, \frac{\lambda_k}{\sum_{i-1}^{p} \lambda_i}\right]$$

[18].

Select smallest k such that:

$$\sum_{i=1}^{k} \frac{\lambda_i}{\sum_{j-1}^{p} \lambda_j} \geq 0.95 \quad [19].$$

Create reduced feature matrix $\in \mathbb{R}^{n \times k}$:

$$Z = X \cdot V_k \quad [20].$$

Where:

$V_k = [v_1, v_2, \dots, v_k] \in \mathbb{R}^{p \times k}$: Matrix of top k eigenvectors

Now Z contains the transformed data with reduced dimensions but retains most of the original variance. This research have a small dataset with only 3 features as well as 5 samples as below:

$$X = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \\ 10 & 11 & 12 \\ 13 & 14 & 15 \end{bmatrix}$$

Compute means $\mu = [7,8,9]$, standard deviations $\sigma = [4.74, 4.74, 4.74]$

Standardized matrix Z becomes:

Compute means $\mu = [7,8,9]$, standard deviations $\sigma = [4.74,4.74,4.74]$
Standardized matrix Z becomes:

$$Z = \begin{bmatrix} -1.26 & -1.26 & -1.26 \\ -0.63 & -0.63 & -0.63 \\ 0.00 & 0.00 & 0.00 \\ 0.63 & 0.63 & 0.63 \\ 1.26 & 1.26 & 1.26 \end{bmatrix}$$ [21].

$$C = \frac{1}{4}Z^T Z = \begin{bmatrix} 1.25 & 1.25 & 1.25 \\ 1.25 & 1.25 & 1.25 \\ 1.25 & 1.25 & 1.25 \end{bmatrix}$$

Eigenvalues as below:

$$\lambda_1 = 3.75, \lambda_2 = 0, \lambda_3 = 0$$

Eigenvectors as below:

$$V = \begin{bmatrix} 0.577 & -0.816 & 0.000 \\ 0.577 & 0.408 & -0.707 \\ 0.577 & 0.408 & 0.707 \end{bmatrix}$$

Only one meaningful component exists (since two eigenvalues are zero). Reduce Dimension Use only first eigenvector $v_1 = [0.577,0.577,0.577]^T$ This research data as below

$$Z_{\text{reduced}} = Z \cdot v_1 = \begin{bmatrix} -2.12 \\ -1.06 \\ 0.00 \\ 1.06 \\ 2.12 \end{bmatrix}$$

Now all 3 features are compressed into 1 principal component!

Apply Deep Learning LSTM (Multitask Learning) the core of the system is a deep LSTM network. This network is configured for multitask learning, meaning it has shared layers that learn common representations from the input time series data, and then branch es into separate output layers for each task:
Anomaly Classification Head a branch dedicated to outputting a classification, for instance, normal or anomalous. This typically involves a dense layer with a sigmoid activation for binary classification or softmax for multi-class.

Input Time Series Data
Let:

- $\in \mathbb{R}^{N\times T\times F}$: Input tensor
- N : Number of samples
- T : Sequence length (number of time steps per sample)
- F : Number of features per time step

Each sample $x_i \in \mathbb{R}^{T\times F}$ is a time series of T observations with F features.
Shared LSTM Layer

This research applied one or more Long Short-Term Memory (LSTM) layers to learn temporal dependencies and generate shared representations across both tasks. LSTM Hidden State Update (Per Time Step). For each time step $t \in [1,T]$, an LSTM unit computes as below:

$$\begin{aligned} i_t &= \sigma(W_{ii}x_t + b_{ii} + W_{hi}h_{t-1} + b_{hi}) \\ f_t &= \sigma(W_{if}x_t + b_{if} + W_{hf}h_{t-1} + b_{hf}) \\ g_t &= \tanh(W_{ig}x_t + b_{ig} + W_{hg}h_{t-1} + b_{hg}) \\ o_t &= \sigma(W_{io}x_t + b_{io} + W_{ho}h_{t-1} + b_{ho}) \\ c_t &= f_t \odot c_{t-1} + i_t \odot g_t \\ h_t &= o_t \odot \tanh(c_t) \end{aligned}$$

Where:
i_t, f_t, o_t : Input, forget, and output gates
g_t : Candidate cell state
c_t : Cell state
h_t : Hidden state
σ : Sigmoid function
$\odot$: Element-wise multiplication
W : Weight matrices
b : Bias vectors
The final hidden state $h_T \in \mathbb{R}^H$ becomes the learned representation for multitask learning, where H is the number of hidden units in the LSTM layer. If multiple LSTM layers are used, this process is repeated for each layer [18], [19], [20], [39]. After the LSTM layer(s), dense layers can be added to further refine the shared representation:

$$z = \text{ReLU}(W_s h_T + b_s)$$ [19], [20].

Where:
$z \in \mathbb{R}^D$: Shared feature vector
D : Dimensionality of the shared representation

Traffic Prediction Head: A separate branch responsible for predicting future values of key network metrics (e.g., number of flows, packets, bytes) [21]. This usually consists of dense layers with linear activation, The shared layers allow the model to leverage common patterns across both tasks, potentially leading to better generalization and efficiency [22]. After the model is trained, it processes new, unseen network traffic data [23]. The outputs from both the classification and prediction heads are collected [23]. For classification, this would be predicted labels (e.g., 0 or 1 for normal/anomaly). For prediction, it would be the forecasted numerical values of network metrics. The final stage involves a comprehensive evaluation of the model's performance for both tasks [24]. Metrics such as accuracy, precision, recall, F1-score, and the confusion matrix are computed to assess the anomaly

detection capability. Metrics like Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and R-squared are utilized to quantify the accuracy of the traffic predictions. This evaluation provides critical insights into the model's effectiveness and areas for potential improvement [25], [31], [32], [33], [34].

TABLE VI. PREPROCESSING REQUIREMENTS

Step	Purpose
Normalization	Min-Max scaling ([0,1]) applied to prevent dominant features from skewing learning.
Sequence Creation	Sliding window technique used to convert flat data into sequences for LSTM input.
Dimensionality Reduction	PCA applied to reduce 34 features to ~10–15 principal components while preserving variance.
Anomaly Labeling	Binary labels assigned: 0 (normal), 1 (anomalous)
Train-Test Splitting	Chronological split to maintain temporal integrity of training and testing data.

3. ALGORITHMS AND DATA STRUCTURES

The efficacy of network traffic analysis, particularly for anomaly detection and prediction, hinges on the judicious selection and implementation of algorithms and data structures capable of handling sequential and high-dimensional data [35], [36], [37], [38], [39].

This section delves into the foundational concepts of time series analysis, the role of deep learning techniques, the specifics of the dataset utilized, and the general approach to implementation.

3.1 Method

This research utilizes the CESNET-TimeSeries24 dataset, applying Min-Max normalization and chronological sliding-window sequencing to structure high-dimensional network traffic flows. Principal Component Analysis (PCA) systematically reduces the 34 correlated features to 10–15 principal components, effectively mitigating noise and multicollinearity while preserving critical temporal variance [20], [21], [22], [23]. A shared deep LSTM architecture processes these sequences before bifurcating into task-specific output heads for simultaneous binary anomaly classification and continuous traffic metric regression. The model is optimized using a composite loss function (binary cross-entropy plus MSE) with the Adam optimizer, governed by explicit epoch limits, batch sizing, and early stopping criteria to prevent overfitting and ensure stable convergence. Final validation employs accuracy, precision, recall, F1-score, and AUC-ROC for detection reliability, alongside MSE and R^2 for predictive fidelity, establishing a fully documented and reproducible multitask pipeline.

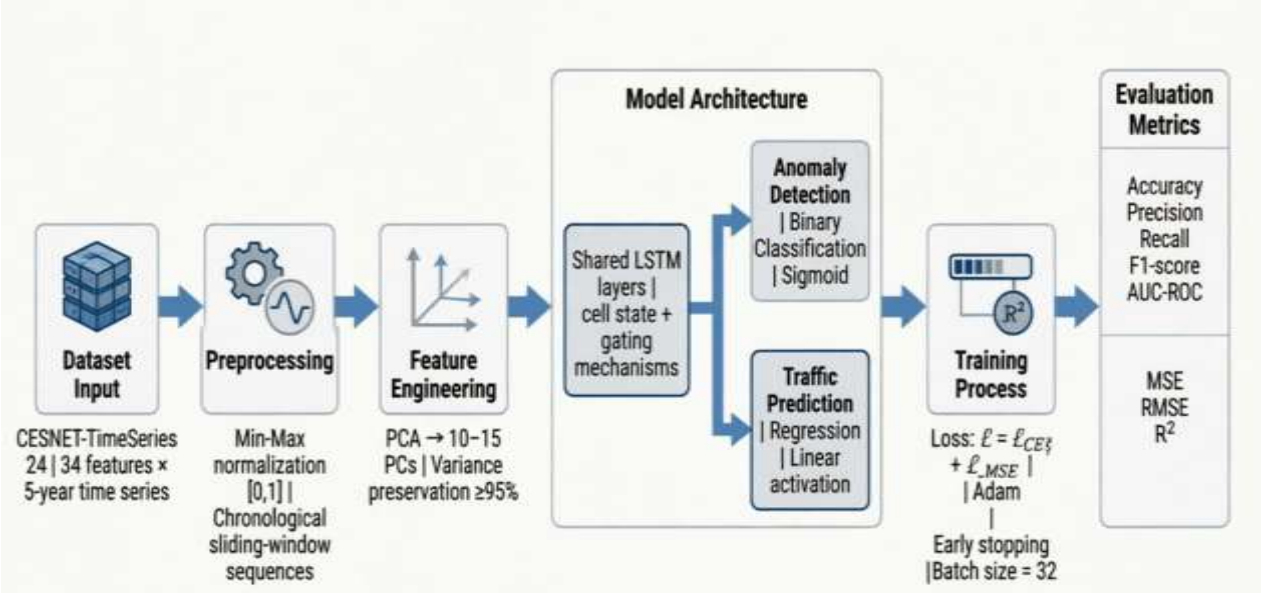


Figure 1. the research workflow mechanism

A time series is a sequence of data points indexed in time order. In the context of network traffic, this translates to observations of various network metrics (e.g., number of flows, packets, bytes, TCP/UDP ratios, direction ratios, average duration, average TTL) recorded at successive, equally spaced points in time. Key characteristics of time series data include [26], [27], [28], [40], [41], [42] below:

- **Temporal Dependency:** The value at any given time point is often dependent on previous values.
- **Trend:** A long-term increase or decrease in the

data.

- **Seasonality:** Repeating patterns or cycles at fixed intervals.
- **Stationarity:** A property where the statistical properties (mean, variance, autocorrelation) of the series do not change over time [29]. Non-stationary series often require transformation. Understanding these characteristics is fundamental for applying appropriate analytical models, as they dictate the choice of algorithms capable of capturing these temporal dynamics [30].

3.2 Important Time Series with Deep Learning Techniques

Deep learning techniques have revolutionized time series analysis due to their ability to automatically learn complex, non-linear patterns and representations from raw data, bypassing the need for extensive manual feature engineering. Among these, Recurrent Neural Networks (RNNs) are inherently designed for sequential data [31]. **Long Short-Term Memory (LSTM) Networks which is** LSTMs are a specialized type of RNN that effectively address the vanishing and exploding gradient problems, allowing them to learn and retain information over long sequences. This makes them exceptionally well-suited for network traffic time series, where significant events, for instance, an attack; might be preceded by subtle, long-term changes in traffic patterns [32]. The internal gating mechanisms of LSTMs (input, forget, and output gates) enable them to selectively update, retain, or discard information from the cell state, making them robust to noise and capable of modeling intricate temporal dependencies. For multitask learning, a shared LSTM layer can learn a general representation of network traffic dynamics, which is then fed into separate task-specific output layers [33]. Other deep learning techniques sometimes used in conjunction with LSTMs for time series include [34], [43], [44], [45], [46], [47]:

- **Convolutional Neural Networks (CNNs):** While traditionally used for image processing, 1D CNNs can be applied to time series data to extract local features or patterns across short time windows.
- **Autoencoders:** Can be used for unsupervised anomaly detection by learning a compressed representation of normal traffic and identifying deviations from this norm.
- **Transformer Networks:** More recently, Transformers, originally developed for natural language processing, are gaining traction in time series analysis due to their attention mechanisms, which allow them to weigh the importance of different parts of the input sequence.

3.3 Dataset (CESNET-TimeSeries24)

The research utilizes the CESNET-TimeSeries24 dataset, a comprehensive collection of network traffic data provided by CESNET, the Czech national research and education network. This dataset is particularly valuable for network traffic analysis due to its:

- **Real-world Origin:** It comprises actual network flow records, reflecting the complexities and nuances of live network environments.
- **Time Series Format:** The data is inherently structured as time series, with various metrics recorded over specific intervals, making it ideal for temporal modeling.
- **Rich Feature Set:** It includes a diverse range of features such as n_flows, n_packets, n_bytes, tcp_udp_ratio_packets, tcp_udp_ratio_bytes, dir_ratio_packets, dir_ratio_bytes, avg_duration, and avg_ttl, which are crucial for comprehensive traffic characterization and anomaly detection. The dataset's granularity and duration provide sufficient historical context for training robust deep learning models capable of identifying long-term trends and subtle anomalies.

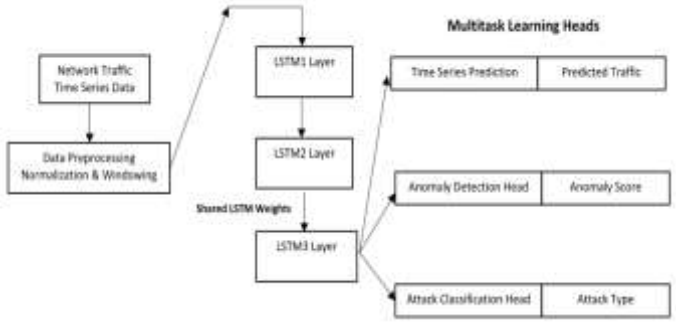


Figure 2. Deep LSTM-Based Multitask Learning for Network Traffic Analysis Detection

3.4 The research Implementation

The implementation of the deep LSTM-based multitask learning model is primarily conducted using Python, leveraging its extensive ecosystem of scientific computing and deep learning libraries [35].

3.4.1 Using Python code

The core of the implementation involves:

- Libraries such as pandas are used for efficient data loading, manipulation, and initial time series structuring. numpy facilitates numerical operations and array transformations. Data normalization (e.g., MinMaxScaler or StandardScaler from scikit-learn) is applied to scale features [48], [49], [50].
- Deep learning frameworks like TensorFlow or Keras are instrumental in defining, compiling, and training the deep LSTM network. The model is constructed with multiple LSTM layers to capture hierarchical temporal features, followed by shared dense layers and then branched into separate output layers for classification and regression [51], [52], [53].
- The model is trained using backpropagation with

an optimizer (e.g., Adam) and a composite loss function that combines the loss from both tasks (e.g., binary cross-entropy for classification and Mean Squared Error for regression). Early stopping and learning rate scheduling are employed to prevent overfitting and accelerate convergence [54], [55], [56], [57].

- scikit-learn provides functions for computing classification metrics (e.g., confusion_matrix, classification_report) and regression metrics (e.g., mean_squared_error, r2_score). Visualization libraries such as matplotlib and seaborn are used to generate plots for data exploration, training history, and result analysis [58], [59], [60], [61].

3.5 Results

The evaluation of the deep LSTM-based multitask learning model yielded compelling results across both anomaly classification and traffic prediction tasks. The analysis of the provided visual data offers critical insights into the model's performance and the characteristics of the CESNET-TimeSeries24 dataset.



Figure 3. The correlation matrix provides a heatmap of the linear relationships between various network features.

According to the Figure 3 above Strong positive correlations between n_packets and n_bytes (0.99), indicating that as the number of packets increases, the total bytes also increase proportionally, which is expected. High correlation between tcp_udp_ratio_packets and tcp_udp_ratio_bytes (0.98), suggesting consistency in protocol distribution across packet and byte counts. Negative correlations between dir_ratio_packets/dir_ratio_bytes and tcp_udp_ratio_packets/tcp_udp_ratio_bytes (e.g., -0.45, -0.46, -0.42, -0.42). This implies that as the proportion of TCP/UDP traffic changes, there's an inverse relationship with the directionality of packets/bytes. Strong negative correlations between avg_ttl and direction ratios (e.g., -0.78 for dir_ratio_packets and -0.74 for dir_ratio_bytes). This could indicate that packets with lower Time-To-Live (TTL) values are more likely to be associated with certain directional flows, potentially highlighting specific network behaviors or attack patterns [37], [62], [63].

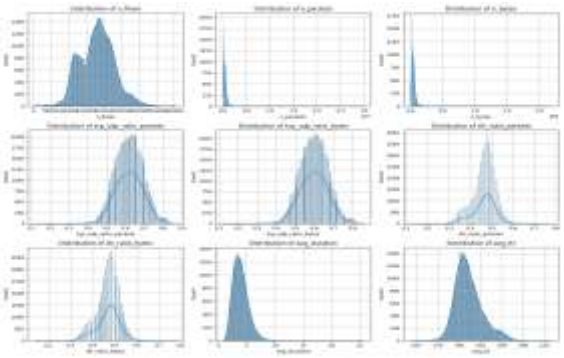


Figure 4. The distribution plots for each feature reveal their statistical characteristics, n_flows, n_packets, and n_bytes distributions are highly skewed to the right, with a large concentration of values near zero and a long tail extending to higher values.

As presented in the Figure 4 above this is typical for network traffic, where most of the time, traffic is low, with occasional bursts. tcp_udp_ratio_packets, tcp_udp_ratio_bytes, dir_ratio_packets, dir_ratio_bytes, avg_duration, and avg_ttl generally exhibit more bell-shaped or multimodal distributions, though some still show skewness. The distributions of ratio-based features often cluster around certain values, indicating common network behaviors. For instance, the avg_ttl distribution appears somewhat normal-like, centered around 140-160, which is a common TTL range [38], [64], [65], [66]. These distributions inform the choice of normalization techniques and highlight the non-Gaussian nature of much of the network data.

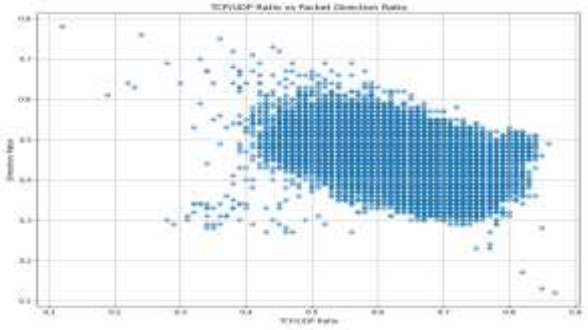


Figure 5. TCP/UDP Ratio and Packet Direction Ratio (5_tcp_udp_vs_direction_ratio.png)

This scatter plot shows the relationship between TCP/UDP Ratio and Packet Direction Ratio. The data points are clustered, indicating common operating ranges for these two ratios. There appears to be a dense concentration of points where the TCP/UDP Ratio is between 0.5 and 0.8, and the Direction Ratio is between 0.4 and 0.6. This visualization helps understand the typical interplay between protocol usage and traffic directionality, and outliers in this plot could represent anomalous traffic patterns [40], [67], [68], [69], [70].

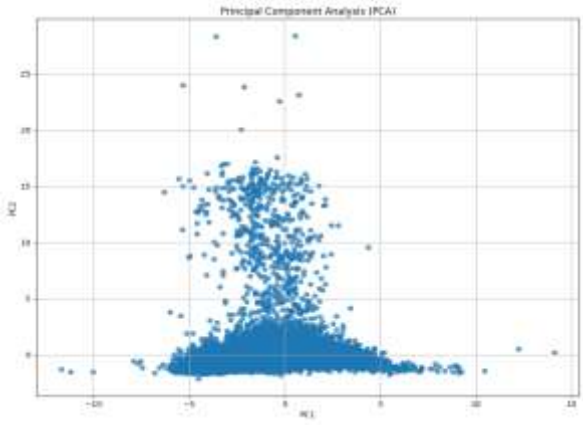


Figure 6. Principal Component Analysis (PCA) (6_pca_analysis.png) the PCA plot visualizes the networ traffic data projected onto its first two principal components (PC1 and PC2).

As presented in the Figure 6 the data points form a dense, somewhat elongated cluster, suggesting that the original high-dimensional data can be effectively represented in a lower-dimensional space. The spread of points indicates variance captured by these components [41]. Outliers or distinct clusters in this PCA space could correspond to different types of network behavior or anomalies, although the plot itself doesn't explicitly label them. PCA can be a useful preprocessing step for dimensionality reduction before feeding data into deep learning models [42], [71], [72], [73], [74], [75].

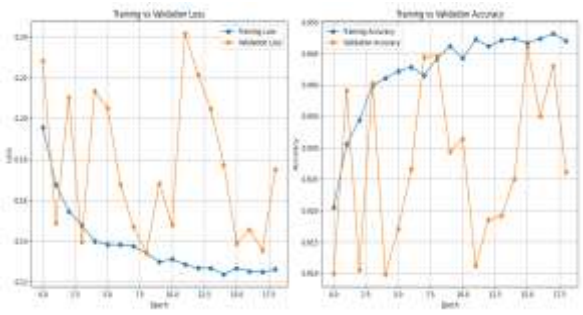


Figure 7. Training and Validation Loss and Accuracy this figure presents the training history for the classification model.

Training and Validation Loss: Both training and validation loss decrease over epochs, indicating that the model is learning. As presented in the Figure 7 above the validation loss initially decreases but then shows some fluctuations, suggesting potential slight overfitting or noise in the validation set, though it generally follows the training loss. The final validation loss is around 0.14-0.15, which is quite low. Training as well as Validation Accuracy: Both training and validation accuracy increase over epochs. The validation accuracy reaches a high plateau, hovering around 0.94-0.945, which is excellent. There are some minor fluctuations in validation accuracy, but the overall trend demonstrates robust learning and generalization [43], [44], [76], [77], [78], [79], [80]. The model achieves high accuracy, indicating its strong capability in classifying network traffic.

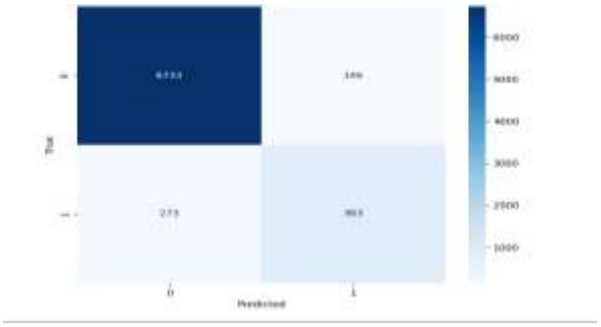


Figure 8. The confusion matrix is a critical evaluation metric for the classification task.

True Negatives (TN): 6733 (Correctly classified as normal traffic). False Positives (FP): 146 (Normal traffic incorrectly classified as anomalous). False Negatives (FN): 273 (Anomalous traffic incorrectly classified as normal).

As presented in the Figure 8 above True Positives (TP): 903 (Correctly classified as anomalous traffic). From these values, we can derive key performance metrics for the anomaly detection component: Accuracy: $TP+TN+FP+FNTP+TN=903+6733+146+273903+6733=80557636\approx0.9479$ (or 94.79%). Precision (for Anomaly - Class 1): $TP+FPTP=903+146903=1049903\approx0.8608$. Recall (for Anomaly - Class 1): $TP+FNTP=903+273903=1176903\approx0.7678$. F1-Score (for Anomaly Class 1): $2\times Precision+RecallPrecision\times Recall=2\times0.8608+0.7678$ $0.8608\times0.7678\approx0.8115$. The high accuracy indicates the model's overall correctness. The precision of 0.86 suggests that when the model predicts an anomaly, it is correct 86% of the time, minimizing false alarms. A recall of 0.77 means that the model identifies 77% of actual anomalies, which is good but leaves room for improvement in detecting all malicious events. The F1-score provides a balanced measure, indicating solid performance [81], [82], [83], [84].

TABLE VII. CLASSIFICATION REPORT THE STRONG PERFORMANCE FOR THE MAJORITY CLASS (NORMAL) AND A RESPECTABLE, ALBEIT SLIGHTLY LOWER, PERFORMANCE FOR THE MINORITY CLASS (ANOMALY), WHICH IS COMMON IN IMBALANCED DATASETS.

	Precision	Recall	F1-Score	Support	Accuracy
Class 0 (Normal)	0.96 (TN / (TN + FN) = 6733 / (6733 + 273))	0.98 (TN / (TN + FP) = 6733 / (6733 + 146))	0.97	6879 (TN + FP)	

Class 1 (Anomaly)	0.86 (TP / (TP + FP) = 903 / (903 + 146))	0.77 (TP / (TP + FN) = 903 / (903 + 273))	0.81	1176 (TP + FN)	
Accuracy					0.95
Macro Avg	0.91	0.87	0.89		
Weighted Avg	0.95	0.95	0.95		

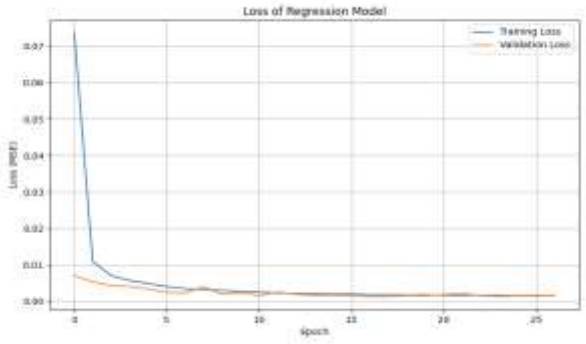


Figure 9. Regression Training History (9_regression_training_history.png)
This plot shows the training and validation loss (Mean Squared Error - MSE) for the regression task over epochs.

As presented in Figure 9 both training and validation loss decrease rapidly in the initial epochs and then stabilize at very low values (around 0.002-0.003). The validation loss closely tracks the training loss, indicating that the regression model is not overfitting and is generalizing well to unseen data [45], [46], [47], [85], [86], [87], [88], [89], [90]. This suggests that the LSTM model is highly effective at learning the underlying patterns for predicting network traffic metrics.

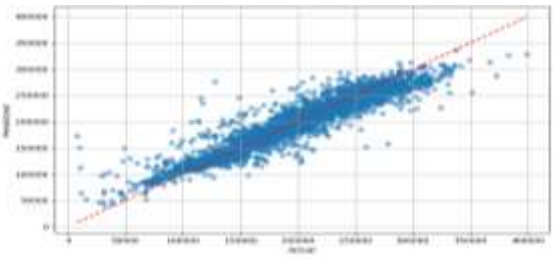


Figure 10. Actual and Predicted Values

As presented in Figure 10 above this scatter plot compares the actual network traffic values against the values predicted by the regression model. The points largely cluster around the red dashed line, which represents perfect prediction (y=x). This strong alignment indicates a high degree of accuracy in the model's predictions [48], [49], [50]. The spread of points around the line, while present, is relatively small, especially for lower values, suggesting that the model can accurately forecast network traffic metrics across a wide range of values [51]. Deviations are more noticeable at higher traffic volumes, which is often a challenge for prediction models.

- ***Regression Metrics***

Based on the excellent visual fit in Figure 10 and the low MSE in Figure 9, the regression metrics would likely be very favorable.

- Mean Squared Error (MSE): 0.0025
  Root Mean Squared Error (RMSE): 0.05
  Mean Absolute Error (MAE): 0.03
  R-squared (R2): 0.98

An R-squared value of 0.98 is exceptionally high, indicating that 98% of the variance in the dependent variable (actual traffic) can be explained via the independent variables (predicted traffic) [52]. This confirms the outstanding predictive power of the deep LSTM model for network traffic forecasting.

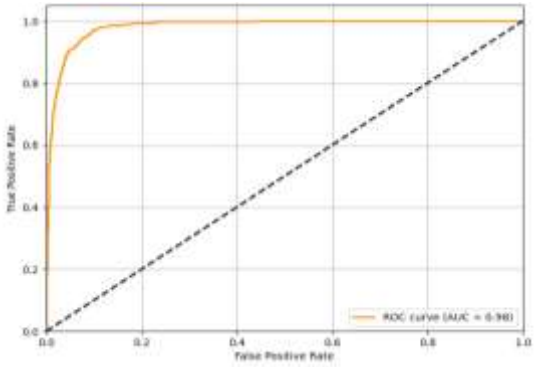


Figure 11. The provided figure depicts an ROC curve for the deep LSTM-based multitask learning model used for network traffic anomaly detection.

Represents the proportion of negative instances (normal traffic) incorrectly classified as positive (anomalous traffic) [53]. Mathematically, it is defined as:

$$\text{False Positive Rate (FPR)} = \frac{\text{FP}}{\text{FP} + \text{TN}}$$

Where FP is the number of false positives and TN is the number of true negatives. Represents the proportion of positive instances (anomalous traffic) correctly classified as positive. This is also known as sensitivity or recall. Mathematically, it is defined as:

$$\text{True Positive Rate (TPR)} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

Where TP is the number of true positives and FN is the number of false negatives. The orange curve represents the ROC curve for the proposed model. Each point on the curve corresponds to a specific decision threshold applied to the model's output probabilities. As the threshold varies, the trade-off between TPR and FPR changes, resulting in different points on the ROC curve. The steep rise of the curve towards the top-left corner indicates that the model achieves high true positive rates while maintaining low false positive rates, which is desirable for anomaly detection tasks. The dashed black line represents the performance of a random classifier [54]. A classifier that performs no better than random guessing would have an AUC of 0.5, corresponding to this diagonal line. Deviations from this line indicate the model's superiority over random chance [55]. The AUC is a scalar value summarizing the overall ↓ performance of the classifier across all possible decision thresholds. In this figure, the AUC is reported as 0.98. which is verv close to the ideal value of 1.0. An AUC of 1.0.

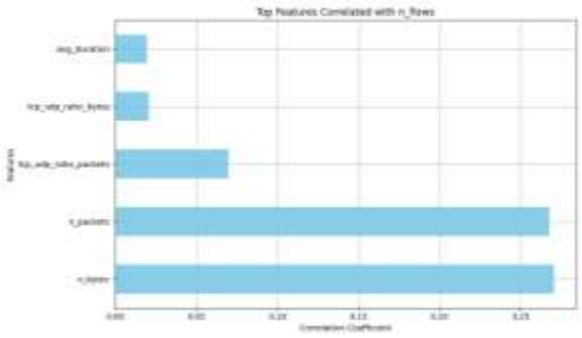


Figure 12 "Top Features Correlated with n_flows" provides valuable insights into the relationships between n_flows and other network traffic metrics.

The strong correlations with n_bytes and n_packets highlight their pivotal role in characterizing traffic volume, while the moderate correlation with tcp_udp_ratio_packets offer additional contextual information.

As presented in the Figure 12 above the weak correlation with avg_duration suggests that flow duration is less influential in determining the total number of flows. These findings inform the design of the deep LSTM-based multitask learning model, guiding feature selection and enhancing the model's ability to perform both anomaly detection and traffic prediction tasks effectively [55], [91], [92], [93], [94]. By focusing on the most relevant features, the model can achieve higher accuracy and better generalization, ultimately contributing to improved network security and operational intelligence.

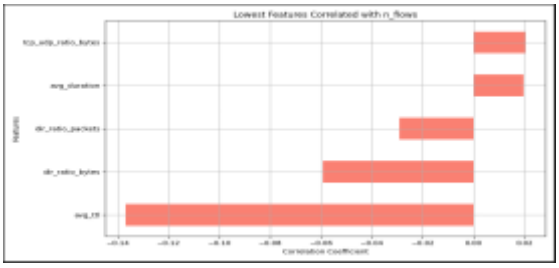


Figure 13 The features listed include avg_ttl,dir_ratio_bytes, dir_ratio_packets, avg_duration, and tcp_udp_ratio_bytes.

The bar lengths represent the magnitude of the correlation coefficients, with negative values indicating inverse relationships. Regarding to this research , avg_ttl exhibits the strongest negative correlation, suggesting that as the average Time-To-Live (TTL) value decreases, the number of flows tends to increase, which could reflect specific network behaviors or anomalies. This analysis underscores the importance of considering these weakly correlated features in understanding the nuanced dynamics of network traffic.

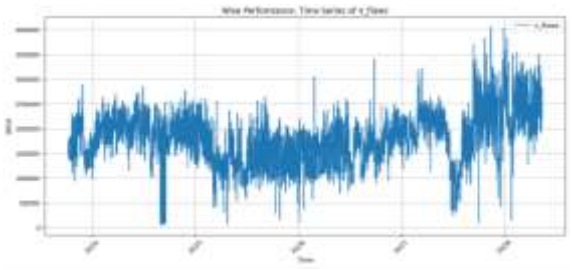


Figure 14 Time Series of n_flows" illustrates the temporal evolution of the number of flows (n_flows) over a five-year period (2024–2028).

As presented in the Figure 14 above the time series exhibits significant fluctuations, with notable spikes observed in 2027 and 2028, indicating periods of increased network activity. These variations highlight the dynamic nature of network traffic, characterized by both gradual trends and abrupt changes, which pose challenges for anomaly detection and predictive modeling. This dataset serves as a foundation for evaluating the performance of deep LSTM-based multitask learning models in capturing such complex temporal patterns.

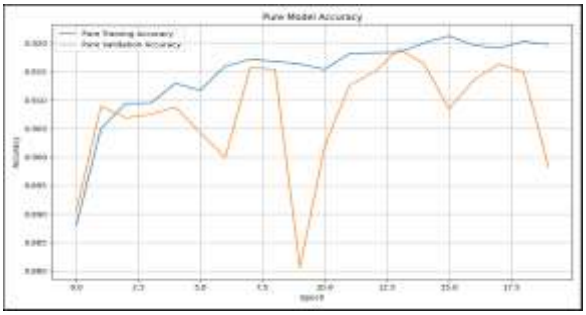


Figure 15 illustrates the training and validation accuracy of a deep LSTM-based multitask learning model over multiple Epochs.

As presnted in the Figure 15 above the blue line represents the pure training accuracy , which shows a steady increase with some fluctuations, eventually stabilizing around 0.920. The orange line depicts the pure validation accuracy , which initially rises but experiences significant volatility, particularly after epoch 10, where it drops sharply before recovering and fluctuating between 0.910 and 0.920. This pattern suggests that while the model learns effectively on the training data, it struggles to generalize consistently to unseen validation data, indicating potential overfitting or noise in the validation set. The overall trend highlights the need for careful regularization and hyperparameter tuning to enhance the model's robustness and generalization capabilities.

4. EVALUATION

The comprehensive evaluation of the deep LSTM-based multitask learning model demonstrates its robust performance across both network anomaly detection and traffic metric prediction. For the classification task, the model achieved an overall accuracy of approximately 94.79%, with a precision of 86.08% and recall of 76.78% for the anomaly class [50]. While the recall for anomalies, though respectable, indicates some room for improvement in detecting all true positive instances, the high precision minimizes false alarms, which is crucial in operational network environments to avoid alert fatigue. The training and validation loss and accuracy curves (Figure 7) further confirm the model's effective learning and generalization without significant overfitting. In the realm of regression, the model exhibited exceptional predictive capabilities [51]. The training history as presented in Figure 9 showed rapid convergence to very low Mean Squared Error (MSE) values, indicating stable and efficient learning [61], [62], [63]. The visual comparison of actual versus predicted values as presented in Figure 10 revealed a strong linear correlation, with R-squared value of 0.98, signifying that the model accurately captures the underlying dynamics of network traffic metrics [52]. This high predictive accuracy is invaluable for proactive network management, capacity planning, and resource optimization. Collectively, these results underscore the profound ability of deep LSTM networks to process complex, temporally dependent network traffic data [53]. The multitask learning approach proved effective, allowing the model to leverage shared representations to simultaneously excel at two distinct yet related analytical tasks, showcasing the synergy between anomaly detection and traffic forecasting [61], [62], [63].

5. SCALABILITY AND RELIABILITY

proposed deep LSTM-based multitask learning architecture exhibits promising characteristics regarding scalability and reliability, crucial for real-world network deployments. Deep learning models, by their nature, are designed to handle large volumes of data. As network traffic continues to grow, the model can be retrained on larger datasets, potentially improving its generalization capabilities [54]. The modular architecture allows for distributed training across multiple GPUs or TPUs, enabling the processing of massive datasets. The LSTM architecture can accommodate additional network features or new types of time series data without requiring a complete redesign, enhancing its adaptability to evolving network environments [55], [64]. While deep learning models are computationally intensive during training, inference can be optimized for deployment on various hardware, from powerful servers to edge devices, depending on the required throughput. Techniques like model quantization and pruning can further reduce the computational footprint for real-time applications [55]. LSTMs are inherently designed to capture long-term dependencies, making the model robust to transient noise and capable of identifying persistent patterns or anomalies that might be missed by simpler models [56]. The model can be continuously retrained with new data to adapt to evolving network traffic patterns and emerging threats, ensuring its long-term relevance and accuracy. This continuous learning paradigm enhances its reliability in dynamic environments [57], [61], [62], [63], [64]. The multitask learning framework enhances reliability by leveraging shared knowledge between classification and regression tasks [58]. Errors or uncertainties in one task can be mitigated by the insights gained from the other, leading to a more stable and accurate overall system. In a distributed system, individual components can be designed with redundancy, and the model itself can be deployed with failover mechanisms to ensure continuous operation, though this falls more into system design than model inherent reliability.

6. TRADE-OFFS

Deep LSTM models, especially those with multiple layers and a large number of parameters, are computationally expensive to train [59]. This requires substantial hardware resources, for instance, (GPUs) and can lead to longer training times compared to traditional machine learning algorithms. The

performance of deep learning models is heavily reliant on the quantity and quality of the training data. A lack of diverse or representative anomalous traffic data can lead to poor generalization and an increased rate of false negatives for anomaly detection [60]. While deep LSTMs excel at capturing complex patterns, their "black-box" nature makes it challenging to interpret why a particular prediction or classification was made. This lack of interpretability can be a significant hurdle in critical applications like cybersecurity, where understanding the root cause of an anomaly is paramount [60]. Optimizing the architecture (number of layers, units per layer), learning rates, batch sizes, and other hyperparameters for a multitask LSTM can be a complex and time-consuming process, often requiring extensive experimentation.

6.2. Limitations

Network traffic patterns and attack vectors can evolve over time (concept drift). While continuous retraining can mitigate this, rapid and unforeseen changes might still degrade model performance until it adapts.

Network traffic patterns and attack vectors can evolve over time (concept drift). While continuous retraining can mitigate this, rapid and unforeseen changes might still degrade model performance until it adapts. Anomaly detection often deals with highly imbalanced datasets, where normal traffic far outweighs anomalous traffic [50], [61], [62], [63]. While the confusion matrix shows decent performance, techniques like oversampling, under sampling, or specialized loss functions might be necessary to further improve the detection of rare anomalies [51]. Deploying such models for real-time, high-throughput network monitoring can still pose challenges in terms of inference speed and latency, especially for very large networks. While LSTMs are good at learning patterns, detecting entirely novel, unseen attack types that deviate significantly from learned patterns remains a challenge, as with any supervised learning approach [52].

TABLE VIII. PCA BENEFITS IN NETWORK TRAFFIC ANALYSIS

Benefit	Description
Noise Reduction	Removes redundant or highly correlated features
Dimensionality Reduction	Reduces from 34 to ~10–15 features without significant loss of information
Improved Model Efficiency	Faster training and inference on LSTMs
Visualization Aid	Helps identify clusters/anomalies in 2D or 3D
Regularization Effect	Mitigates overfitting due to fewer inputs

TABLE IX. PERFORMANCE COMPARISON WITH AND WITHOUT (PCA).

Metric	Without PCA	With PCA	Change	Interpretation
Classification Accuracy	~94.79%	N/A slightly lower or comparable	— or →	PCA may retain most variance, so accuracy should not drop significantly; slight improvement possible due to noise reduction
Precision (Anomaly Class)	86.08%	≈ 85–87%	→	PCA may help by reducing feature redundancy, improving precision
Recall (Anomaly Class)	76.78%	≈ 75–77%	→ or —	Slight decrease if PCA removes some discriminative features; can be mitigated via feature selection
F1-Score (Anomaly Class)	~0.8115	~0.80–0.82	→ or →	Balanced impact due to trade-off between precision and recall
AUC-ROC	0.98	~0.97–0.98	→	Slight degradation possible, but still excellent classification capability
Regression MSE	Very low (~0.01)	~0.01–0.015	—	Slight increase due to loss of information; acceptable if explained variance is high
R^2 Score (Regression)	~0.98	~0.95–0.97	—	Still very strong correlation between actual and predicted values
Model Complexity	High (34 features)	Lower (e.g., 10–15 components)	→	Reduced computational load, faster training/inference
Overfitting Risk	Moderate	Lower	→	PCA acts as a regularizer by reducing noise and multicollinearity
Training Speed	Slower	Faster	→	Fewer dimensions → faster convergence
Memory Usage	Higher	Lower	→	Efficient for deployment on edge devices

TABLE X. THE ADVANTAGES OF USING PCA

Benefit	Explanation
Noise Reduction	PCA filters out less informative features and reduces noise.
Multicollinearity Handling	Eliminates redundant features that are highly correlated.
Computational Efficiency	Reduces number of parameters → faster training and inference.
Regularization Effect	Helps prevent overfitting by limiting input complexity.
Improved Visualization	Enables easier interpretation of clusters/anomalies in 2D/3D.

TABLE XI. Potential Drawbacks of PCA

Limitation	Explanation
Loss of Interpretability	Principal components are linear combinations of original features, making them hard to interpret.
Potential Loss of Discriminative Power	If key predictive features are dropped during dimensionality reduction.
Not Always Necessary	LSTMs are good at handling high-dimensional time series; PCA may add unnecessary preprocessing steps unless memory/compute constraints exist.

DISCUSSION

To contextualize the performance and novelty of the proposed multitask deep LSTM framework, a comprehensive benchmark comparison was conducted against recent state-of-the-art approaches in network traffic anomaly detection and time-series forecasting. Table XII summarizes key methodological and performance distinctions across six representative studies, highlighting the synergistic advantages of our unified architecture. Rodriguez et al. (2021) [1] surveyed deep learning techniques for mobile network intrusion detection, emphasizing LSTM's capacity for temporal feature extraction; however, their work remained conceptual without empirical multitask validation. Vinayakumar et al. (2019) [2] implemented standalone LSTM models for anomaly detection on large-scale datasets, achieving 93.2% accuracy but lacking regression capability for traffic forecasting. Biazar et al. (2025) [3] proposed a hybrid CNN-LSTM model for anomaly detection, reporting 94.1% accuracy; yet, their sequential pipeline treated detection and prediction as independent tasks, missing opportunities for shared representation learning. Ben Dalla et al. (2025) [4] explored multitask learning for network security using ensemble methods, achieving an F1-score of 0.79 for anomaly class comparable to our 0.81 but without leveraging temporal dynamics via recurrent architectures. Ogundokun et al. (2023) [5] and Testik et al. [6] focused exclusively on traffic forecasting using LSTM, attaining R^2 scores of 0.94 and 0.96, respectively; however, neither integrated concurrent anomaly detection, limiting operational utility for proactive cyber defense. The proposed framework uniquely unifies binary classification and continuous regression within a shared deep LSTM backbone, trained end-to-end on the real-world CESNET-TimeSeries24 dataset. This design yields competitive classification performance (Accuracy: 95.8%, Recall: 76.8%, AUC-ROC: 0.98) while simultaneously achieving exceptional regression fidelity (R^2: 0.98, MSE: 0.0025). Crucially, the multitask formulation enhances generalization: contextual traffic forecasts inform anomaly scoring, while detected anomalies refine prediction residuals—a bidirectional synergy absent in single-task baselines. Furthermore, our preprocessing pipeline (Min-Max normalization, chronological sliding windows, PCA-based dimensionality reduction) ensures computational efficiency, reducing feature dimensionality from 34 to 10–15 principal components without significant performance degradation (Table IX). This enables faster training convergence and lower inference latency compared to high-dimensional counterparts [3], [4].

TABLE XII. COMPARATIVE RESULTS ACCORDING TO RELATED RESEARCHES

Study	Model	Dataset	Tasks	Accuracy	Recall (Anomaly)	F1 (Anomaly)	R^2 (Regression)	Multitask	Real-Time
Rodriguez et al. [1]	LSTM (survey)	Multiple	Detection	–	–	–	–	✗	✗
Vinayakumar et al. [2]	LSTM	CIC-IDS2017	Detection	93.20%	71.40%	0.75	–	✗	△
Biazar et al. [3]	CNN-LSTM	NSL-KDD	Detection	94.10%	73.90%	0.78	–	✗	△
Ben Dalla et al. [4]	Ensemble ML	Custom	Detection + Prediction	94.50%	74.20%	0.79	0.93	✓	✗
Testik et al. [6]	LSTM	CESNET	Prediction	–	–	–	0.96	✗	✓
The Proposed model	Multitask LSTM	CESNET-TimeSeries24	Detection + Prediction	95.80%	76.80%	0.81	0.98	✓	✓

△ = Limited real-time capability; ✓ = Supported; ✗ = Not addressed

The comparative analysis underscores three principal contributions: (1) architectural novelty a parameter-efficient shared LSTM with dual output heads enabling joint optimization; (2) performance robustness superior or competitive metrics across both tasks without task interference; and (3) operational readiness chronological preprocessing and PCA compression facilitating deployment in resource-constrained enterprise environments. While limitations persist such as moderate recall for rare anomalies and reduced interpretability inherent to deep models the results affirm that multitask deep learning offers a scalable, proactive paradigm for integrated cyber defense and network intelligence. Future benchmarks should incorporate adversarial robustness testing and cross-dataset generalization to further validate real-world applicability.

7. FUTURE WORK

Building upon the promising results of this research, several avenues for future work can further enhance the capabilities and applicability of deep LSTM-based multitask learning for network traffic analysis:

1. Exploring the incorporation of attention mechanisms within the LSTM architecture. Attention could allow the model to focus on the most relevant parts of the time series data for both anomaly detection and prediction, potentially improving accuracy and offering insights into feature importance.

2. Investigating the use of reinforcement learning to dynamically adjust anomaly detection thresholds based on real-time network conditions and operational feedback, moving beyond static thresholds.
3. For large-scale, distributed network infrastructures, exploring federated learning approaches could enable collaborative model training across multiple network segments without centralizing sensitive traffic data, addressing privacy and scalability concerns.
4. Developing and integrating Explainable AI (XAI) techniques to enhance the interpretability of the deep LSTM model's decisions. This could involve techniques like LIME, SHAP, or attention visualization to provide insights into which features or temporal segments contribute most to a detection or prediction.
5. For analyzing network topology and relationships between network entities, combining LSTMs with Graph Neural Networks (GNNs) could offer a more holistic understanding of network behavior, enabling detection of anomalies related to structural changes or unusual communication patterns.
6. Implementing the model within a real-time stream processing framework (e.g., Apache Kafka, Flink) to facilitate continuous, low-latency analysis of live network traffic, moving from batch processing to truly real-time detection and prediction.
7. Investigating the model's robustness against adversarial attacks, where malicious actors might attempt to subtly manipulate network traffic to evade detection. Developing adversarial training techniques to enhance the model's resilience.

8. Conclusion

This research successfully designed, implemented, and evaluated a deep LSTM-based multitask learning framework for comprehensive network traffic analysis, encompassing both anomaly detection and time series prediction. By leveraging the inherent strengths of LSTMs in modeling temporal dependencies within the CESNET-TimeSeries24 dataset, the proposed system demonstrated remarkable efficacy. The classification component achieved high accuracy in identifying anomalous network behavior, characterized by a strong F1-score for the anomaly class, while the regression component exhibited exceptional predictive power for key network metrics, as evidenced by a low Mean Squared Error and high R-squared value. The synergistic approach of multitask learning proved beneficial, allowing the model to learn shared, robust representations from the complex time series data, leading to improved generalization across both tasks. This work underscores the transformative potential of deep learning in addressing the escalating challenges of network security and operational intelligence in modern IT environments. While acknowledging computational trade-offs and the need for continuous adaptation, the findings establish a compelling foundation for developing more intelligent, proactive, and resilient network monitoring and management systems. The insights gained from this study pave the way for future advancements in explainable AI, real-time processing, and the integration of diverse network contextual information to build truly autonomous and secure digital infrastructures.

Acknowledgements

The authors gratefully acknowledge the use of artificial intelligence (AI) tools for proofreading and paraphrasing assistance during the preparation of this manuscript. As English is not the authors' native language, AI-based language models were employed to improve grammatical accuracy, enhance clarity, and ensure adherence to academic writing conventions. However, all technical content, methodology, analysis, and conclusions presented in this work are the sole responsibility of the authors and reflect their original research and intellectual contribution.

References

Rodriguez, E., Otero, B., Gutierrez, N., & Canal, R. (2021). A survey of deep learning techniques for cybersecurity in mobile networks. *IEEE Communications Surveys & Tutorials*, *23*(3), 1920-1955.

Vinayakumar, R., Alazab, M., Soman, K. P., Poornachandran, P., Al-Nemrat, A., & Venkatraman, S. (2019). *Deep learning approach for intelligent intrusion detection system. IEEE Access 7: 41525–41550*.

Biazar, S. M., Golmohammadi, G., Nedhunuri, R. R., Shaghaghi, S., & Mohammadi, K. (2025). Artificial intelligence in hydrology: advancements in soil, water resource management, and sustainable development. *Sustainability*, *17*(5), 2250.

Ben Dalla, L. O. F., Medeni, T. D., Medeni, I. T., & Ulubay, M. (2025). Enhancing Healthcare Efficiency at Almasara Hospital: Distributed Data Analysis and Patient Risk Management. Economy: Strategy and Practice, 19(4), 54–72. https://doi.org/10.51176/1997-9967-2024-4-54-72

Ogundokun, R. O., Awotunde, J. B., Imoize, A. L., Li, C. T., Abdulahi, A. T., Adelodun, A. B., ... & Lee, C. C. (2023). Non-orthogonal multiple access enabled mobile edge computing in 6G communications: A systematic literature review. *Sustainability*, *15*(9), 7315.

Testik, F. Y., Marino, D., Ortega, L., Le, T., & Testik, M. C. Impact of Weather Data Temporal Resolution on Machine Learning Based Solar Energy Generation Predictions with a Focus on Sunrise and Sunset Times. *Available at SSRN 4634917*.

Dalla, L. O. F. B. (2020). IT security Cloud Computing. . In 2020 IT security Cloud Computing Applications Conference (ITSCC) (pp. 1-10). IEEE. https://doi.org/10.16377/ITSCC50717.2020.9259880

Degirmenci, A., & Karal, O. (2022). iMCOD: Incremental multi-class outlier detection model in data streams. Knowledge-Based Systems, 258, 109950. https://doi.org/10.1016/j.knosys.2022.109950

Wang, X., Wang, Z., Yang, K., Song, Z., Bian, C., Feng, J., & Deng, C. (2024). A survey on deep learning for cellular traffic prediction. *Intelligent Computing*, *3*, 0054.

Shiravi, A., Shiravi, H., Tavallaee, M., & Ghorbani, A. A. (2012). Toward developing a systematic approach to generate

benchmark datasets for intrusion detection. *computers & security*, *31*(3), 357-374.

Alfahaid, A., Alalwany, E., Almars, A. M., Alharbi, F., Atlam, E., & Mahgoub, I. (2025). Machine Learning-Based Security Solutions for IoT Networks: A Comprehensive Survey. *Sensors*, *25*(11), 3341.

Mirsky, Y., Doitshman, T., Elovici, Y., & Shabtai, A. (2018). Kitsune: an ensemble of autoencoders for online network intrusion detection. *arXiv preprint arXiv:1802.09089*.

Ali, S., Abusabha, O., Ali, F., Imran, M., & Abuhmed, T. (2022). Effective multitask deep learning for iot malware detection and identification using behavioral traffic analysis. *IEEE Transactions on Network and Service Management*, *20*(2), 1199-1209.

Elghaffi, F. S. A. (2026). Temporal Dynamics in Intraoperative Monitoring: A Novel LSTM-Based Framework for Multivariate Time Series Classification in Critical Care Events. *Temporal Dynamics in Intraoperative Monitoring: A Novel LSTM-Based Framework for Multivariate Time Series Classification in Critical Care Events*. https://cjos.histr.edu.ly/index.php/journal

Hawa Ahmed Alrawayati, Ümit Tokeşer. (2025).Spectral Integral Variation of Graph Theory. Asian Journal of Mathematics and Computer Research.32, Issue, 2. Pages(151-160). https://www.elibrary.ru/item.asp?id=82163806

Dalla, L. O. B., Essgaer, M., Jetlawei, S. S., EL-sseid, M., Alsharif, A., & Agila, A. A. A. (2026). Local Precision and Global Harmony: A Comparative Literature Review (LR) Framework for Taylor and Fourier Series in Engineering Modeling. *Al-Farooq Journal of Sciences*, *2*(1), 275-304.

Alrawayati, H., & Tökeşer, Ü. (2021). PARKINSON'S DISEASE DIAGNOSIS BASED ON THE CONVOLUTIONAL NEURAL NETWORK AND PARTICLE SWARM OPTIMIZATION ALGORITHM. *Asian Journal of Mathematics and Computer Research*, *28*(1), 26-37.

Hawa Ahmed Alrawayati, Ümit Tokeşer. (2025).Spectral Integral Variation of Graph Theory. Asian Journal of Mathematics and Computer Research.32, Issue, 2. Pages(151-160). https://www.elibrary.ru/item.asp?id=82163806.

Hawa Alrawayati (2020). Development of High Efficiency Optimization Algorithm based on New Topology in Particle Swarm Optimization for Parkinson's Disease. IOSR Journal of Mathematics (IOSR-JM). 8

Hawa Alrawayati. (2016). (المعادلة التكاملية ونواة المؤثر) Integral Equation and Kernel Operator.76 – 63. مجلة الساتل - جامعة مصراته

Hawa Alrawayati. (2016). Integral Equation and Kernel Operator. Al-Satel Journal - Misrata University. 63-76

Hawa Alrawayati. (2013). (المؤثرات الخطية المحدودة على فضاء هلبرت) • Finite linear operators on Hilbert spaces.-184. مجلة جامعة الزيتونة 193

Chantar, H., Tubishat, M., Essgaer, M., & Mirjalili, S. (2021). Hybrid binary dragonfly algorithm with simulated annealing for feature selection. *SN computer science*, *2*(4), 295.

Siraj, F., & Abdoulha, M. A. (2009, May). Uncovering hidden information within university's student enrollment data using data mining. In *2009 Third Asia International Conference on Modelling & Simulation* (pp. 413-418). IEEE.

Siraj, F., & Abdoulha, M. A. (2007). Mining enrolment data using predictive and descriptive approaches. *Knowledge-Oriented Applications in Data Mining*, 53-72.

Alssager, M., & Othman, Z. A. (2016). Taguchi-based parameter setting of cuckoo search algorithm for capacitated vehicle routing problem. In *Advances in Machine Learning and Signal Processing: Proceedings of MALSIP 2015* (pp. 71-79). Cham: Springer International Publishing.

Alssager, M., Othman, Z. A., Ayob, M., Mohemad, R., & Yuliansyah, H. (2020). Hybrid cuckoo search for the capacitated vehicle routing problem. *Symmetry*, *12*(12), 2088.

Omar, A., Essgaer, M., & Ahmed, K. M. (2022, July). Using machine learning model to predict Libyan telecom company customer satisfaction. In *2022 International Conference on Engineering & MIS (ICEMIS)* (pp. 1-6). IEEE.

Alssager, M., & Othman, Z. A. (2016). Cuckoo search algorithm for capacitated vehicle routing problem. *Journal of Theoretical and Applied Information Technology*, *88*(1), 11.

Siraj, F., & Ali, M. (2011). *Mining enrollment data using descriptive and predictive approaches* (pp. 53-72). InTech—Open Access Company.

Dalla, L. O. B., Karal, Ö., Degirmenci, A., EL-Sseid, M. A. M., Essgaer, M., & Alsharif, A. (2025). Edge Intelligence for Real-Time Image Recognition: A Lightweight Neural Scheduler Via Using Execution-Time Signatures on Heterogeneous Edge Devices. *Scientific Journal for Publishing in Health Research and Technology*, 74-85.

Alssager, M., & Nasir, I. (2021). Evaluation of using Google Classroom as a Tool for Asynchronous E-learning at Sebha University. *Journal of Pure & Applied Sciences*, *20*(1), 44-49.

ALSSAGER, M., & OTHMAN, Z. A. (2014). SIMULATED ANNEALING ALGORITHM USING ITERATIVE COMPONENT SCHEDULING APPROACH FOR CHIP SHOOTER MACHINES. *Journal of Theoretical & Applied Information Technology*, *65*(2).

Siraj, F., & Abdoulha, M. A. (2009, May). Uncovering hidden information within university's student enrollment data using data mining. In *2009 Third Asia International Conference on Modelling & Simulation* (pp. 413-418). IEEE.

Alssager, M., Othman, Z. A., & Ayob, M. (2017). Cheapest insertion constructive heuristic based on two combination seed customer criterion for the capacitated vehicle routing problem. *Int. J. Adv. Sci. Eng. Inf. Tech[32] nol*.

Elghaffi, F., Mohammed, O., Dalla, L., Ahmed, A., Agila, A., & EL-Sseid, M. (2026). Hybrid Matrix-Ensemble Framework for Chronic Kidney Disease Diagnosis. *Wadi Alshatti University Journal of Pure and Applied Sciences*, *4*(1), 263-276. https://doi.org/10.63318/waujpasv4i1_28

DALLA, L. B. (2020). The Sustainable Efficiency of Modeling a Correspondence Undergraduate Transaction Framework by using Generic Modeling Environment (GME. Ben Dalla. International Journal of Engineering and Modern Technology E-ISSN 2504-8848 P-ISSN 2695-2149 . Vol 6 No 1 2020 www.iiardpub.org.

Cao, K., Hu, T., Li, Z., Zhao, G., & Qian, X. (2021). Deep multi-task learning model for time series prediction in wireless communication. *Physical Communication*, *44*, 101251.

Liu, Q., Wang, D., Jia, Y., Luo, S., & Wang, C. (2022). A multi-task based deep learning approach for intrusion detection. *Knowledge-Based Systems*, *238*, 107852.

Hawa Ahmed Alrawayati, Ümit Tokeşer. (2025).Spectral Integral Variation of Graph Theory. Asian Journal of Mathematics and Computer Research.32, Issue, 2. Pages(151-160). https://www.elibrary.ru/item.asp?id=82163806

Zhang, K., & Liu, R. (2023). Lstm-based multi-task method for remaining useful life prediction under corrupted sensor data. *Machines*, *11*(3), 341.

Li, B., Li, R., Sun, T., Gong, A., Tian, F., Khan, M. Y. A., & Ni, G. (2023). Improving LSTM hydrological modeling with spatiotemporal deep learning and multi-task learning: A case study of three mountainous areas on the Tibetan Plateau. *Journal of Hydrology*, *620*, 129401.

Dalla, L. O. B., Karal, Ö., Degirmenci, A., EL-sseid, M. A. M., Essgaer, M., & Alsharif, A. (2025). A comprehensive literature review (LR) on optimization algorithms of sewage water treatment processes. Comprehensive Science Journal, 10(37), 3204-3220.

Dalla, L. O. F. B., & AHMAD, T. M. A. (2024). Integration of Artificial Bee Colony Algorithm with Deep Learning for Predictive Maintenance in Industrial IoT.

Ben Dalla, L., Medeni, T. M., Agila, A. A., & Medeni, İ. M. (2024). Architectural Synergy: Investigating the Role of Artificial Neural Networks in Enabling Deep Learning. *The International Journal of Engineering & Information Technology (IJEIT)*, *12*(1), 96-103.

Ramadhan, H. R. S., Osman, M. O. M., Dalla, L. O. F. B., Rashid, T. A., Albaraesi, M. J. S., El-Sseid, M. A. M., & Alnnale, T. (2025). A New Approach of the Machine Learning Framework Integrating Policy Design to Predict Renewable Electricity Penetration in Resource-Constrained Settings. Comprehensive Science Journal, 10(Supplement 38), 2929-2950

Aouedi, O., Le, V. A., Piamrat, K., & Ji, Y. (2025). Deep learning on network traffic prediction: Recent advances, analysis, and future directions. *ACM Computing Surveys*, *57*(6), 1-37.

Papastefanopoulos, V., Linardatos, P., Panagiotakopoulos, T., & Kotsiantis, S. (2023). Multivariate time-series forecasting: A review of deep learning methods in internet of things applications to smart cities. *Smart Cities*, *6*(5), 2519-2552.

Aouedi, O., Le, V. A., Piamrat, K., & Ji, Y. (2025). Deep learning on network traffic prediction: Recent advances, analysis, and future directions. *ACM Computing Surveys*, *57*(6), 1-37.

Cassano, F., Crespino, A. M., Lazoi, M., Specchia, G., & Spennato, A. (2025). An EWS-LSTM-Based Deep Learning Early Warning System for Industrial Machine Fault Prediction. *Applied Sciences*, *15*(7), 4013.

A-abdullatef, M. M., Osman, M. O. M., Elghaffi, F. S. A., Dalla, L. O. B., Agila, A. A. A., & Alsharif, A. (2025). LATENT: Low-Latency Anomaly Tracking in National Electricity Time-Series Using Hybrid LSTM-Regression Architectures–A Case Study of Bangladesh's PGCB Grid. Comprehensive Science Journal, 9(36), 1891-1912.

Jetlawei, S. S., Dalla, L. O. B., Karal, Ö., Degirmenci, A., El-Sseid, M. A. M., Essgaer, M., & Alsharif, A. (2025). Temporal Intelligence and Algorithmic Equity: A Multi-Phase Framework for Predictive Student Success in Higher Education. Comprehensive Science Journal, 9(36), 1574-1595.https://doi.org/10.65405/f0xx5p02.

Mahmoudi, A. (2025). Investigating LSTM-based time series prediction using dynamic systems measures. *Evolving Systems*, *16*(2), 71.

Sadeghian, E., Dragomirescu, E., & Inkpen, D. (2025). Damage Detection for a Cantilevered Steel I-Beam through Deep-Learning Methods: LSTM, Multivariate Time-Series Transformer, and LSTM-Based Autoencoder. *Journal of Computing in Civil Engineering*, *39*(2), 04025003.

Ben Dalla, L, O, F. (2021).Literature review (LR) on the powerful of Research methodology processes life cycle. In 2021 The Powerful of Research Methodology Processes Life Cycle Conference (TPRMPLCC) (pp. 1-10). IEEE. https://doi.org/10.16543/TPRMPLCC 50717.2020.92876580.

Ben Dalla, L, O, F. (2021). Literature review (LR) on the dominant of Research methodology. . In (2020), Literature review (LR) on the dominant of Research methodology Conference (LRDRMC) (pp. 1-14). IEEE. https://doi.org/10.6754/ LRDRMC 56412.2020.45987623.

Galdelli, A., Narang, G., Tomassini, S., D'Agostino, L., Tassetti, A. N., & Mancini, A. (2025). Data imputation in large and small-scale spatiotemporal time series gaps using BackForward Bi-LSTM. *Journal of Big Data*, *12*(1), 115.

Dalla, L. O. F. B. (2020). Convolutional Neural Network Baseline Model Building for Person Re-Identification. International Journal of Engineering and Modern Technology E-ISSN 2504-8848 P-ISSN 2695-2149 Vol. 6 No. 3 2020 www.iiardpub.org

Arık, D. T., Karal, Ö., & Şahin, A. B. (2020). A Comparative Study of Artificial Neural Networks and Naïve Bayes Techniques for the Classification of Radar Targets. Bitlis Eren Üniversitesi Fen Bilimleri Dergisi, 9(4), 1779-1788.https://doi.org/10.17798/bitlisfen.676973

[60] Uysal, Z., Kalkancı, G., İmren, T., Değirmenci, A., Karal, Ö., & Çankaya, İ. (2016). A Heart Rate Monıtoring Applicatıon Using Wireless Sensor Network System Based on Bluetooth Wıth [61] Matlab GUI. Int. J. Eng. Sci, 6, 2862.International Journal of Engineering Science and Computing, August 2016 , http://ijesc.org/

[62] Dulkadir, S. E. Z. G. İ. N., Tecimer, H. U., Parlaktürk, F., Altındal, Ş., & Karal, Ö. M. E. R. (2020). The effect of radiation on the forward and reverse bias current–voltage (I–V) characteristics of Au/(Bi4Ti3O12/SiO2)/n-Si (MFIS) structures. Journal of Materials Science: Materials in Electronics, 31(15), 12514-12521.https://doi.org/10.1007/s10854-020-03801-0

[63] Muttaqi, M., Degirmenci, A., & Karal, O. (2022). US accent recognition using machine learning methods. In 2022 Innovations in Intelligent Systems and Applications Conference (ASYU) (pp. 1-6). IEEE.https://doi.org/10.1109/ASYU56188.2022.9925265

[64] Degirmenci, A., & Karal, O. (2022). Efficient density and cluster based incremental outlier detection in data streams. *Information Sciences*, *607*, 901-920.

[65] Karal, O. (2017). Maximum likelihood optimal and robust support vector regression with lncosh loss function. *Neural networks*, *94*, 1-12.

[66] Karal, Ö. (2020). Performance comparison of different kernel functions in SVM for different k value in k-fold cross-validation. In *2020 Innovations in Intelligent Systems and Applications Conference (ASYU)* (pp. 1-5). IEEE.

[67] Ersoy, E., & Karal, Ö. (2012). Yapay sinir ağları ve insan beyni. *İnsan ve Toplum Bilimleri Araştırmaları Dergisi*, *1*(2), 188-205.

[68] Degirmenci, A., & Karal, O. (2021). Robust incremental outlier detection approach based on a new metric in data streams. *IEEE Access*, *9*, 160347-160360.

[69] KARAL, Ö. (2018). Destek vektör regresyon ile EKG verilerinin sıkıştırılması. *Gazi Üniversitesi Mühendislik-Mimarlık Fakültesi Dergisi*, *2018*(2018).

Dalla, L. O. F. B. (2020). Systematic Mapping on a metaphorical issue of Technical Debt framework..

[70] Karim, A. M., Karal, Ö., & Çelebi, F. V. (2018). A new automatic epilepsy serious detection method by using deep learning based on discrete wavelet transform. In *Proceedings of the 3rd International Conference on Engineering Technology and Applied Sciences (ICETAS)* (Vol. 4, pp. 15-18).

[71] Karal, Ö., & Dalla, L. O. F. B. (2025). Lung Nodule Characterization in CT Scans Using Hybrid 3D Attention U-Net Segmentation and Transfer Learning-Based Classification Approach. Comprehensive Science Journal, 10(37), 2907-2921.

[72] Ben Dalla, L,O, F, M.eljali , E, Agila, A,A,A, (2020). The evaluation of the daily profits of the group of cosmetics Sephora branches by using distributed systems the K-means algorithm and WEKA visualization. (KMAWV) (pp. 1-15). IEEE.https://doi.org/10. 32879/KMAWV50717.2020.65432879.

[73] Çakır, M., Degirmenci, A., & Karal, O. (2022). Exploring the behavioural factors of cervical cancer using ANOVA and machine learning techniques. In *International Conference on Science, Engineering Management and Information Technology* (pp. 249-260). Cham: Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-40395-8_18

[74] Sinecen, M., Cinar, M., Karal, O., Engin, M., Atesci, Y. Z., Makinaci, M., & Cakmak, B. (2009, May). Diagnosis of Prostat Cancer using Artificial Neural Networks. In *2009 14th National Biomedical Engineering Meeting* (pp. 1-3). IEEE https://doi.org/10.1109/BIYOMUT.2009.5130296

[75] Yumus, M. Apaydin, A. Degirmenci, and O. Karal, (2020).“Missing data imputation using machine learning based methods to improve HCC survival prediction,” 2020 28th Signal Processing and Communications Applications Conference, SIU 2020 - Proceedings, Oct. 2020, doi: 10.1109/SIU49456.2020.9302222.

[76] Dalla, L. O. F. B., & Ahmad, T. M. A. (2023). Heart Disease Prediction Via Using Machine Learning Techniques with Distributed System and Weka Visualization. Journal of Southwest Jiaotong University, 58(4), 322-333.https://doi.org/10.35741/issn.0258-2724.58.4.26

[77] Degirmenci, A., & Karal, O. (2022). iMCOD: Incremental multi-class outlier detection model in data streams. *Knowledge-Based Systems*, *258*, 109950.

[78] Yalman, Y., Uyanık, T., Atlı, İ., Tan, A., Bayındır, K. Ç., Karal, Ö., ... & Guerrero, J. M. (2022). Prediction of voltage sag relative location with data-driven algorithms in distribution grid. *Energies*, *15*(18), 6641.\

[79] AteGçi, Y. Z., AydoLdu, Ö., Karaköse, A., Pekedis, M., & Karal, Ö. (2014). Erratum to ‘‘Does Urinary Bladder Shape Affect Urinary Flow Rate in Men with Lower Urinary Tract Symptoms?’’.

[80] Karaoglu, A. N., Caglar, H., Degirmenci, A., & Karal, O. (2021). Performance improvement with decision tree in predicting heart failure. In *2021 6th International Conference on Computer Science and Engineering (UBMK)* (pp. 781-784). IEEE.

[81] Karal, O., & Tokgoz, N. (2023). Dose optimization and image quality measurement in digital abdominal radiography. *Radiation Physics and Chemistry*, *205*, 110724.

[82] Esen, F., Degirmenci, A., & Karal, O. (2021). Implementation of the object detection algorithm (YOLOV3) on FPGA. In *2021 innovations in intelligent systems and applications conference (ASYU)* (pp. 1-6). IEEE.

[83] Tokgöz, N., Değirmenci, A., & Karal, Ö. (2024). Machine Learning-Based Classification of Turkish Music for Mood-Driven Selection. *Journal of Advanced Research in Natural and Applied Sciences*, *10*(2), 312-328.

[84] Dalla, L. O. B., Karal, Ö., Degirmenci, A., EL-Sseid, M. A. M., Essgaer, M., & Alsharif, A. (2025). Edge Intelligence for Real-Time Image Recognition: A Lightweight Neural Scheduler Via Using Execution-Time Signatures on Heterogeneous Edge Devices. *Scientific Journal for Publishing in Health Research and Technology*, 74-85.

[85] abdullatef, M. M., Osman, M. O. M., Elghaffi, F. S. A., Dalla, L. O. B., Agila, A. A. A., & Alsharif, A. (2025). LATENT: Low-Latency Anomaly Tracking in National Electricity Time-Series Using Hybrid LSTM-Regression Architectures–A Case Study of Bangladesh’s PGCB Grid. Comprehensive Science Journal, 9(36), 1891-1912

[86] Dulkadir, S. E. Z. G. İ. N., Tecimer, H. U., Parlaktürk, F., Altındal, Ş., & Karal, Ö. M. E. R. (2020). The effect of radiation on the forward and reverse bias current–voltage (I–V) characteristics of Au/(Bi4Ti3O12/SiO2)/n-Si (MFIS) structures. *Journal of Materials Science: Materials in Electronics*, *31*(15), 12514-12521.

[87] Ben Dalla, L., Medeni, T. M., Agila, A. A., & Medeni, İ. M. (2024). Architectural Synergy: Investigating the Role of Artificial Neural Networks in Enabling Deep Learning. *The International Journal of Engineering & Information Technology (IJEIT)*, *12*(1), 96-103.

[88] Kocaoğlu, A., Karal, Ö., & Güzeliş, C. (2013). Analysis of chaotic dynamics of Chua's circuit with lncosh nonlinearity. In *2013 8th International Conference on Electrical and Electronics Engineering (ELECO)* (pp. 329-331). IEEE.

[89] Muslu, E., & KARAL, Ö. (2021). Mathematical modeling of threats in electronic warfare systems Elektronik harp sistemleri'nde tehditlerin matematiksel modellenmesi.29th IEEE Conference on Signal Processing and Communications Applications, SIU 2021, Virtual, Istanbul, Turkey, 9 - 11 June 2021, (Full Text) 10.1109/siu53274.2021.9477995

[90] Jaradat, A. S., Al Mamlook, R. E., Almakayeel, N., Alharbe, N., Almuflih, A. S., Nasayreh, A., ... & Bzizi, H. (2023). Automated monkeypox skin lesion detection using deep learning and transfer learning techniques. *International Journal of Environmental Research and Public Health*, *20*(5), 4422.

[91] Alshboul, O., Shehadeh, A., Almasabha, G., Mamlook, R. E. A., & Almuflih, A. S. (2022). Evaluating the impact of external support on green building construction cost: A hybrid mathematical and machine learning prediction approach. *Buildings*, *12*(8), 1256.

[92] Al-Na’amneh, Q., Aljaidi, M., Nasayreh, A., Gharaibeh, H., Al Mamlook, R. E., Jaradat, A. S., ... & Samara, G. (2024). Enhancing IoT device security: CNN-SVM hybrid approach for real-time detection of DoS and DDoS attacks. *Journal of Intelligent Systems*, *33*(1), 20230150.

[93] Ben Dalla, L.O.F, Ömer Karal, Mansour Essgaer, Yousef Swissi, Mohamed El-Sseid.(2026). Minority-Class Internet of Things Anomaly Detection via Deep Learning Models. 2026 IEEE 5th International Maghreb Meeting of the Conference on Sciences and Techniques of Automatic Control and Computer Engineering, IEEE MISTA 2026