International Journal of Safety and Security Engineering
Vol. 16, No. 4, April, 2026, pp. 913-921
Journal homepage: http://iieta.org/journals/ijsse

A Trust-Enhanced TabTransformer Framework for Intrusion Detection in Software-Defined Networks

Manar H. Bashaa[1,2*], Wesam S. Bhaya[1], Nabeel H. Kaghed Al-aaraji[1]

[1] College of Information Technology, University of Babylon, Babil 51002, Iraq
[2] College of Computer Science and Information Technology, University of Kerbala, Kerbala 56001, Iraq

Corresponding Author Email: manar.h@uokerbala.edu.iq

https://doi.org/10.18280/ijsse.160418

Received: 19 February 2026
Revised: 12 April 2026
Accepted: 24 April 2026
Available online: 30 April 2026

Keywords:
Software-Defined Networking, Trust_Score, intrusion detection systems, TabTransformer, machine learning, deep learning

ABSTRACT

Software-Defined Networking (SDN) provides greater flexibility and centralized control over the network, but also leads to a higher level of complexity for intrusion detection because of heterogeneous and dynamic aggregation of traffic. Even though deep-learning-based intrusion detection systems (IDSs) have shown high accuracy, the majority of work to date is either a black-box or ignores the different trustworthiness of network flows. In order to overcome this limitation, this paper introduces a Trust-Enhanced TabTransformer (TE2T) that can directly support multi-class intrusion detection in SDN environments. The proposed framework, which combines Trust Score, computed through Random Forest, into a trust-aware attention mechanism. TE2T is evaluated over three benchmark datasets, CICIDS2017, INSDN, and UNSW-NB15, with traditional and SDN-native traffic. The experimental results indicate that TE2T delivers strong multi-class detection capability with 97.98% accuracy and 99.42 ROC-AUC on CICIDS2017, 97.91% accuracy and 98.54 ROC-AUC on INSDN, and 99.27% accuracy on UNSW-NB15 in highly imbalanced and complex traffic conditions. Using state-of-the-art IDS approaches as comparison points from recent literature, the results show that TE2T achieves competitive or superior performance in terms of accuracy and detection speed, yet has a simpler architecture and enhanced robustness compared to the comparison points, thus making it appropriate for practical SDN security deployments.

1. INTRODUCTION

By the decoupling of the control plane from the data plane, Software-Defined Networking (SDN) represents a novel paradigm in modern network management, characterized by facilitating programmability, flexibility, and centralized oversight of network resources [1-3]. While this architectural flexibility can facilitate dynamic configuration and efficient traffic handling, it also comes with new security challenges. This makes the centralized controller the brain of the SDN itself the obviously high-value target to cyber attackers, and exposes the network to types of potential threats, namely spoofing, Denial of Service (DoS), Distributed DoS (DDoS), and flow table overloading attacks [4, 5]. Therefore, it is important to have a good and smart intrusion detection systems (IDSs) in order to keep the correctness and trust of SDN infrastructures infrastructures [6, 7].

Traditional IDSs that depend on shallow machine-learning models usually fail to generalize through heterogeneous network environments due to their heavy reliance on manual feature selection and handcrafted rules [8]. Deep learning has significantly improved the accuracy of intrusion detection through the automated learning of latent patterns in network traffic; however, these models still behave as a black box without interpretability and robustness to noisy or uncertain data. Additionally, they do not consider the trust of flows which may emerge as an important attribute in security for SDN, as controllers combines traffic from many different and possibly untrustable sources [9, 10].

In order to get over these challenges, this research proposes a novel Trust-Enhanced TabTransformer (TE2T) framework for SDN intrusion detection. The proposed model builds upon the TabTransformer architecture, which efficiently handles mixed categorical and numerical network features using contextual embedding's and self-attention mechanisms [11, 12]. The Trust Score is a unique part of the proposed approach. It is calculated using a Random Forest model that looks at how well each flow behaves based on how its features are related to attack labels. The proposed trust-guided attention mechanism is integrated in the Transformer encoder, thus enabling the model to automatically push high-confidence traffic while suppressing flow from low-confidence or adversarial flows during feature learning.

To further confirm the adaptability and generalization capacity of the suggested framework under various network conditions, the proposed framework is evaluated on three SDN-related benchmark datasets UNSW-NB15, CICIDS2017, and INSDN. The experimental results demonstrate that the TE2T outperforms multiple state-of-the-art deep learning models with respect to detection accuracy, precision, recall

and f1.

This work's primary contributions are outlined as follows:

(1) In this work, we propose a framework named Trust-TE2T that incorporates trust-guided attention into a welladopted TabTransformer architecture for SDN intrusion detection.
(2) In this work correlation-based filtering, recursive feature elimination (RFE), and Random Forest-based Trust Score are included to improve feature reliability through hybrid feature selection approach.
(3) In particular, we propose a novel trust-guided attention mechanism that explicitly integrates instance-level reliability into the attention computation, enabling reliability-aware feature interaction.
(4) Extensive experiments on three SDN datasets show competitive superior performance and cross-dataset generalization.

2. RELATED WORKS

In recent years, numerous deep learning-based IDSs have been developed for SDN. Sequential-based models, such as RNN, LSTM, and GRU, have been extensively used to capture temporal dependencies in network traffic. For instance, Tang et al. [13] introduced DeepIDS, a deep neural framework utilizing DNN and GRU architectures to examine OpenFlow traffic characteristics derived from the NSL-KDD dataset. Their investigations revealed that the GRU-RNN model attained almost 89% accuracy and F1-score, surpassing the DNN variant (nearly 81%) and conventional classifiers including SVM, Decision Tree, and Naïve Bayes. When deployed on a POX controller, the system showed nearly 4% reduction in performance with minimal overheads as the number of switches increased, thus confirming the feasibility of practical usage of DL-based IDS in real SDN systems.

Chaganti et al. [14] suggested a deep learning-based IDSs for SDN-IoT environments, utilizing Long Short-Term Memory (LSTM) networks to classify assaults in flow-level traffic gathered via OpenFlow. Their research evaluated various machine learning and deep learning algorithms, such as DNN, SVM, CNN, and LSTM, utilizing two datasets produced in actual SDN-IoT testbeds (DS1 and DS2). The proposed four-layer LSTM architecture achieved 97.1% accuracy and 0.99 AUC on multiclass attack detection, outperforming traditional machine learning and other deep learning models. The model was able to detect DoS as well as fuzzing, DDoS port scanning, and OS fingerprinting attacks and was sufficiently generalized to give a fair detection on independent datasets.

Besides sequential models, hybrid and advanced deep learning models have also been studied to enhance the performance of detection and tackle issues like class imbalance. Zhang et al. [15] proposed TIBS, an advanced deep-learning intrusion detection framework combining Transformer, Inception, BiGRU, and self-attention modules for SDN environments. To mitigate class imbalance, the authors integrated an improved Auxiliary Classifier GAN (ACGAN) employing residual learning and BiGRU-based dynamic sampling, generating balanced training data for model optimization. Evaluated on the CIC-IDS-2017 and CIC-DDoS-2019 datasets, TIBS achieved 95.66% accuracy (F1 = 0.83) and 96.48% accuracy (F1 = 0.90), outperforming existing CNN-Transformer and RNN-based approaches.

To improve the performance of intrusion detection in SDN setting, hybrid and advanced deep learning methods are also suggested. Agrawal et al. [16] Proposed a DRL + GCN framework to enhance intrusion detection in SDN. Their hybrid design takes advantage of the adaptive nature of policy updating of DRL to keep track of the changes in threat landscape but uses GCNs to model graph-structured SDN traffic and learn topological dependencies among flows. Evaluated on the NSL-KDD dataset, the system attains 95.85% accuracy and an F1-score of 96.87 %, significantly outperforming conventional deep models such as DNN and GRU-RNN.

Bose et al. [17] introduced a multi-layered security framework for intrusion detection in SDN environments, combining classical machine-learning and deep-learning techniques. Their BAT-MC model employed multiple convolutional layers, Bidirectional LSTM, and an attention mechanism to automatically extract spatial-temporal patterns in network traffic, eliminating the need for manual feature engineering. Evaluated on the In-SDN dataset, which includes diverse attack categories (DoS, DDoS, phishing, MITM, SQL injection, and brute-force), the BAT-MC achieved 86 % accuracy. Although the literature concentrates on sequential, convolutional, or hybrid architectures, they do not directly include attention mechanisms that are trust-aware as in this work.

3. BACKGROUND

This section provides a brief summary of the Transformer architecture and its adaptations for tabular data in the TabTransformer paradigm.

3.1 Transformer architecture

Transformer were proposed by Vaswani et al. [18]. Fully attention-based sequence modeling replaced all recurrent structures with fully attention-based ones. In contrast to RNN or LSTM models that process input in a sequential manner, Transformers can compute in parallel through self-attention which allows the model to naturally grasp long-range dependencies and contextual relationships further advanced than prior models. A traditional transformer in Figure 1 is made up of an encoder and a decoder [19].

The encoder receives an input sequence and transforms it into a set of continuous representations through stacked layers of multi-head self-attention and feed-forward networks. The decoder, used primarily in tasks like translation or text generation, takes these encoded representations and produces the target sequence. Each attention layer computes the relationship between input tokens using three key matrices: Query (Q), Key (K), and Value (V). The self-attention mechanism measures how much attention each element in the input should pay to others using the following function:

$$Attention(Q,K,V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \quad (1)$$

where, d_k denotes the dimensionality of the keys, and the softmax normalization ensures that attention weights sum to one. By using multi-head attention, the Transformer allows the model to attend to information from different representation subspaces simultaneously, capturing diverse relationships

among features. To maintain positional information (since attention lacks an inherent sense of order), positional encoding is added to the input embedding's, enabling the model to differentiate the position of each token [18, 19].

The encoder output is then passed through feed-forward layers and normalized via layer normalization to stabilize training. Due to its parallelism, scalability, and ability to model complex contextual dependencies, the Transformer has become the backbone for many advanced architectures including BERT, GPT, and Tab Transformer and has proven highly effective in capturing relationships in diverse data types such as text, images, and, more recently, tabular network traffic [18, 19].

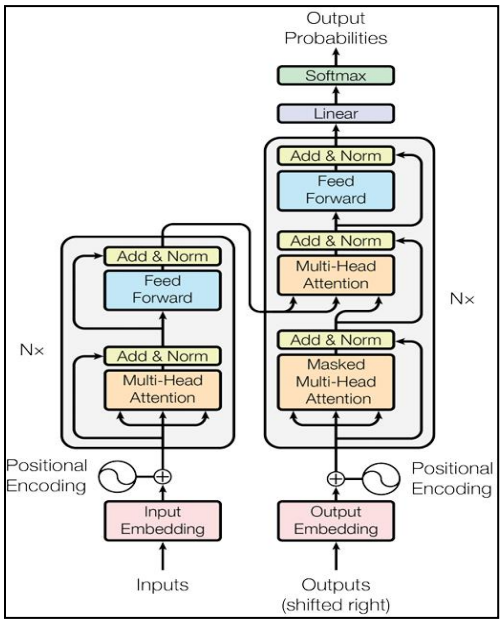


Figure 1. Transformer architecture [18]

3.2 Extension to TabTransformer for tabular Data

The Transformer architecture, at first intended for natural language processing, has been adapted for structured or tabular data. A prominent adaptation is TabTransformer [11] TabTransformer: Tabular Data Modeling Using Contextual Embeddings which integrates self-attention mechanisms for tabular learning tasks. Instead of using sparse one-hot encodings for categorical features, TabTransformer embeds those features into dense vectors, then processes them through multi-head self-attention layers to produce contextual embeddings that capture inter-feature interactions dynamically. These contextualized embeddings are then concatenated with normalized numerical features and fed into downstream prediction networks (e.g., MLP) for classification or regression. Through experiments on 15 tabular benchmark datasets, the authors report that TabTransformer outperforms other deep learning baselines by at least 1.0% in mean AUC and is competitive with tree-ensemble models. Additionally, TabTransformer's representations show robustness to missing or noisy features and offer interpretability via attention weights. Figure 2 illustrates the architecture of TabTransformer [11, 20].

In network traffic analysis and SDN-based intrusion detection, data is tabular by nature, being composed of both categorical (protocol type, port) and numerical (packet counts, durations etc.) flow features. In this respect, deploying TabTransformer in this domain is advantageous since it attention mechanism allows features to be weighted dynamically based on how they interact within a given context; an ability that is particularly crucial for telling whether a flow is benign or malicious [11, 20].

Based on the capabilities of TabTransformer, the proposed Trust-Enhanced Transformer extends TabTransformer by including a trust-guided attention layer, where Trust Scores are included as additional contextual signals into attention calculations. This mechanism enhances detection explainibility and accuracy by enabling the model to pay attention to confident flows and to down-weight uncertain or noisy data, particularly in sophisticated and adversarial SDN environments.

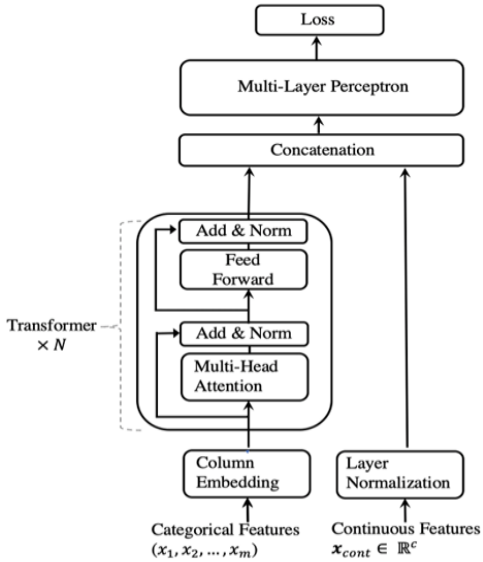


Figure 2. TabTransformer architecture [11]

4. PROPOSED WORK

This research presents an innovative framework referred to as TE2T, which augments the standard TabTransformer architecture with a trust-aware attention mechanism for enhanced interpretability and robustness in SDN intrusion detection. The comprehensive structure of the suggested TE2T framework is clarified in Figure 3. The suggested framework is intended to work within the SDN control plane, in which the trained TE2T model is implemented at the controller level. The data plane is gathered and processed to produce real-time inferences to generate attack-aware decisions. SDN simulators like Mininet and controllers like POX can be used to implement the framework to evaluate it in real-time.

4.1 Trust Score computation

During this stage, a Trust Score indicates how well-behaved each flow is, which creates a metric of reliability. It starts by checking which of the numerical attributes are most informative when taking the attack category into account using correlation analysis. Features 15 most correlated with the behavior of the flow. We then train a Random Forest classifier on these selected features where the target variable separate Normal and Attack traffic. After training, the model will output the probability of the flow belonging to the normal class. Such estimated trust score is between 0–1.

Thus the Trust Score is confidence of classifier regarding real behavior of network as it perceived. Higher Trust Score Flows are required to match the traffic patterns we expect. Low trust score means bot or malicious activity. And, this probabilistic trust estimation can be incorporated into dataset and the quality of the sample can be leveraged in subsequent stages of learning to improvement the robustness and detection performance.

The proposed Trust Score is not a traditional confidence measure but rather an independent estimate of instance-level reliability based on an auxiliary model. It is not applied as a regular feature as with standard methods but subsequently included in the TE2T architecture as a special signal to direct the attention mechanism so that interaction between features is reliably supported.

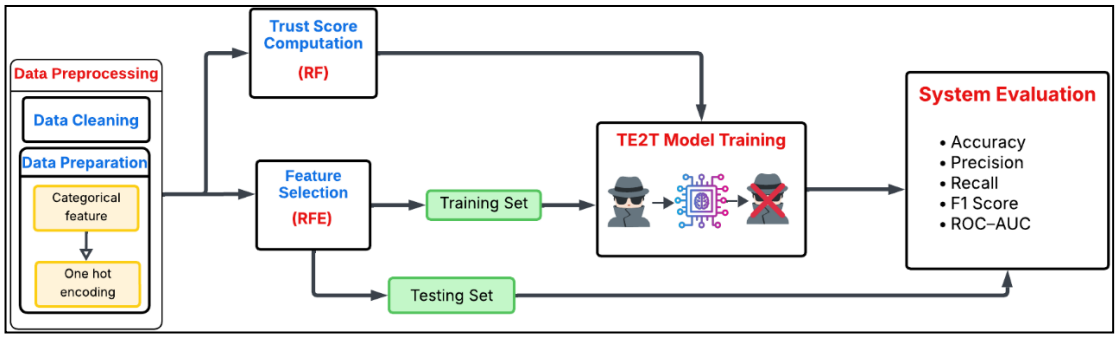


Figure 3. The proposed Trust-Enhanced TabTransformer (TE2T) framework

4.2 Feature selection

Then a hybrid feature selection strategy is applied to refine the input space. The categorical attributes (service,proto, state) are one-hot-encoded and concatenated with numerical features. Next, RFE based on Random Forest is applied on the training data set to obtain 20 most discriminative attributes that classify an attack. The final step is the subset to which trust score calculated in last step is used to derive the feature vector. Such integration guarantees that the chosen features will preserve statistically significant as well as behaviorally dependable features, which progress the robustness and interpretable of the model.

4.3 Trust-Enhanced TabTransformer architecture

TE2T approaches behavioral reliability by incorporating a Trust Score into the standard attention mechanism within TabTransformer. Firstly the categorical features (proto, service, state) are embedded, and the numerical attributes based on RFE, and the computed Trust Score are mapped to a common latent space through a simple linear layer. The mixture of such feature representations are then fed into a stacked TrustAttention, composed of two layers that build over the typical self-attention mechanism by adding the Trust Score as an additional bias term.

Formally, the modified attention is computed as

$$Attention(Q,K,V) = softmax\left(\frac{QK^T}{\sqrt{d_k}} + \alpha \mathrm{T}\right)V \qquad (2)$$

where, T represents the broadcasted Trust Score, the parameter alpha α is a tunable scaling factor controlling the influence of the trust bias, which is added before the softmax operation to modulate attention weights without altering normalization. This formulation allows the model to dynamically change the weights of feature interactions based on how reliable the behavior is, giving more weight to trustworthy flows and less weight to suspicious patterns. The aggregated attention outputs are then processed through a multilayer perceptron (MLP) for multi-class attack detection. By combining statistical discriminability and trust-aware weighting, the proposed architecture achieves enhanced robustness, interpretability, and precision in SDN traffic analysis.

5. EXPERIMENTS

5.1 Datasets

This paper utilized three publicly available and SDN relevant intrusion detection datasets: UNSW-NB15 [21], CICIDS2017 [22], and INSDN [23] for comprehensive evaluation of the proposed TE2T framework. These datasets, when taken together, include a large variety of network behaviors, from generic Internet traffic to SDN-specific attack scenarios, making possible a cross-domain evaluation of the model adaptability and generalization ability.

5.2 Preprocessing

The dataset was preprocessed to fit into deep learning via removing extra spaces, forcing non-numeric values into numeric and filling null values with zeros. For feature selection, categorical attributes such as Protocol and Service were temporarily converted using one-hot encoding to enable the Random Forest-based RFE process. In the deep learning stage, categorical attributes were instead transformed into integer indices using label encoding and fed into embedding layers within the TabTransformer architecture, avoiding one-hot encoding during model training. The data were split into 80% for training and 20% for testing, while 10% of the training set was further reserved for validation to support early stopping.

5.3 Trust Score computation and feature selection

To address reliability, a Trust Score is computed using a Random Forest trained on the top-15 correlated numeric features with the attack label, representing the probability of a flow being normal. Subsequently, Recursive Feature

Elimination (RFE) with a Random Forest estimator selects the top-20 discriminative attributes. The final feature space therefore consists of the RFE-selected attributes augmented with the computed Trust Score. Class imbalance is mitigated through stratified sampling and weighted metrics, without oversampling.

5.4 Model training

The proposed TE2T model was implemented in PyTorch and trained independently on the UNSW-NB15, CICIDS2017, and INSDN datasets to ensure fair cross-domain evaluation. For each dataset, the input feature space combined the most relevant numerical attributes selected through feature selection with the computed Trust Score, while categorical features such as Service, Protocol, or State were encoded through embedding layers instead of sparse one-hot encoding.

Training employed the Adam optimizer [24] with dataset-specific learning rates between 1×10^{-5} and 3×10^{-5}, a batch size of 64, and the cross-entropy loss for multi-class classification. The model was trained for up to 150-200 epochs, using early stopping based on validation loss (patience = 10-15, min-delta = 1×10^{-4}) to prevent overfitting. All experiments performed stratified splits of 80% for training and 20% for testing, while 10% of the training set. The Algorithm 1 demonstrates how the proposed system will work.

5.5 Evaluation metrics

Five standard metrics of Accuracy, Precision, Recall, F1-score, and ROC-AUC are used to evaluate the proposed TE2T model. These quantitative measures are of the ability of the model to classify multiple different attack categories.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{3}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{4}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{5}$$

$$\text{F1 score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{6}$$

where, TP, TN, FP, and FN refers to true positive, true negative, false positive, and false negative, respectively. ROC-AUC (Receiver Operating Characteristic-Area Under Curve) indicates the discriminative power of the model, where higher AUC values imply better separation between normal and attack traffic.

Algorithm 1: TE2T model

Input: Dataset D with features (numerical + categorical) and label column

Output: Trained TE2T model M, selected feature set F, and evaluation metrics.

Step 1: Load and Clean Data D

Step 2: Compute Trust Score

- Create a binary target is_normal: is_normal = 1 if attack_cat == "Normal", else 0.
- Select the top N_{trust} most correlated numerical features → F_{trust} with attack_cat
- Train a Random Forest classifier RF_{trust} on (X_{trust}, y_{trust}).
- For each valid record, compute: Trust_Score = P(normal | record) using predict_proba.
- Append Trust_Score to the dataset and save the updated file.

Step 3: Feature Selection

- Apply RFE to select top N_{rfe} features → F_{rfe}.
- Form the final feature set: F = $F_{rfe} \cup \{\text{Trust\_Score}\}$.

Step 4: Split Dataset (Train / Validation / Test)

- Perform stratified split into training and testing sets (80/20).
- Split a validation subset from the training set (10% of training) using stratification.

Step 5: Train Trust-Enhanced TabTransformer

- Initialize a TabTransformer model with: categorical embeddings, numerical feature projection, Trust Score projection.
- Integrate Trust Score into the attention computation to bias attention weights.

Step 6: Early Stopping

- If validation loss improves by at least MIN_DELTA, save the model as the current best.
- If no improvement occurs for PATIENCE consecutive epochs, stop training.
- Restore the best saved model parameters.

Step 7: Testing and Evaluation

- Compute evaluation metrics: Accuracy, Precision, Recall, F1-score.
- Compute multi-class ROC-AUC when class distribution allows it, and plot ROC curves.

6. RESULTS

As demonstrated in the experimental evaluations, the proposed TE2T achieves competitive and stable performance on the CICIDS2017, INSDN and UNSW-NB15 datasets, which validates its capability to adapt to different network traffics. Table 1 summarizes the performance of the proposed TE2T model based on weighted multi-class evaluation metrics, on the independent test sets.

The model has achieved high accuracy, precision and weighted F1-score on CICIDS2017, affirming its ability to reliably detect volumetric flooding (DDoS, DoS Hulk) and application-layer (web-based) attacks. Combined with the high recall scores, these results reaffirm that the model can detect a wide variety of attacks while providing very few false negatives.

Even for the INSDN dataset, which corresponds to SDN-native traffic samples captured from within a controller environment, the proposed model still achieves high accuracy and F1-score, albeit in a smaller input space. The findings validate that the trust-aware attention mechanism indeed contributes to the robustness improvement in controller-level SDN scenarios.

Similarly, TE2T achieves competitive performance over all attacks on UNSW-NB15 as well, particularly in Exploits, Fuzzers, and Reconnaissance categories. The model still achieves very high accuracy and F1-score, which means that performance in multi-class classification is stable, despite

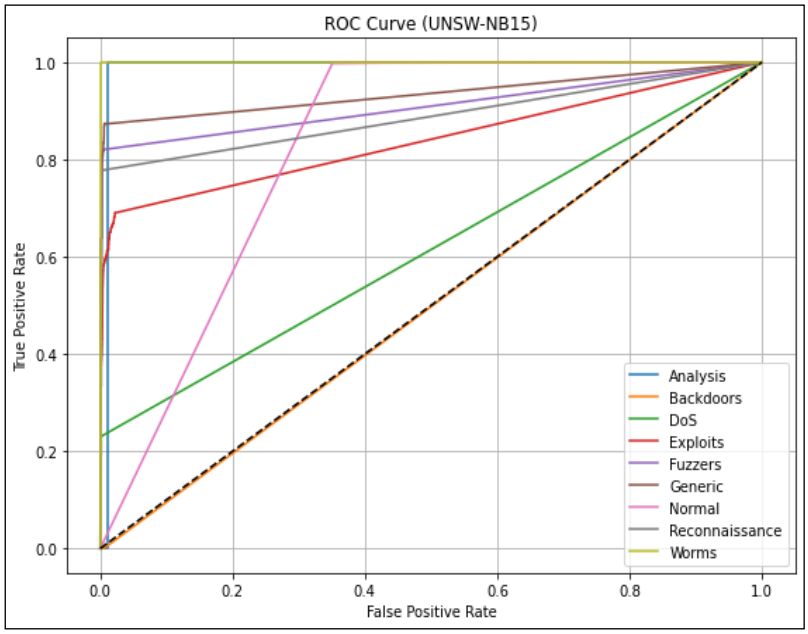
some rare attack classes having a slightly lower recall (due to class-imbalance). The relatively lower ROC-AUC is due to the increased class overlap and the complexity involved in the UNSW-NB15 dataset rather than a degradation in the overall detection performance. This is more pronounced in some classes of attacks where there is a similarity in the traffic, resulting in low separability even though the overall accuracy and F1-score are high.

The high precision values across all data sets suggest a low false-positive rate, which is crucial for practical purpose IDSs. In addition to this, ROC-AUC results shown in Figure 4 further reveal good class separability and generalizability to SDN specific network environments.

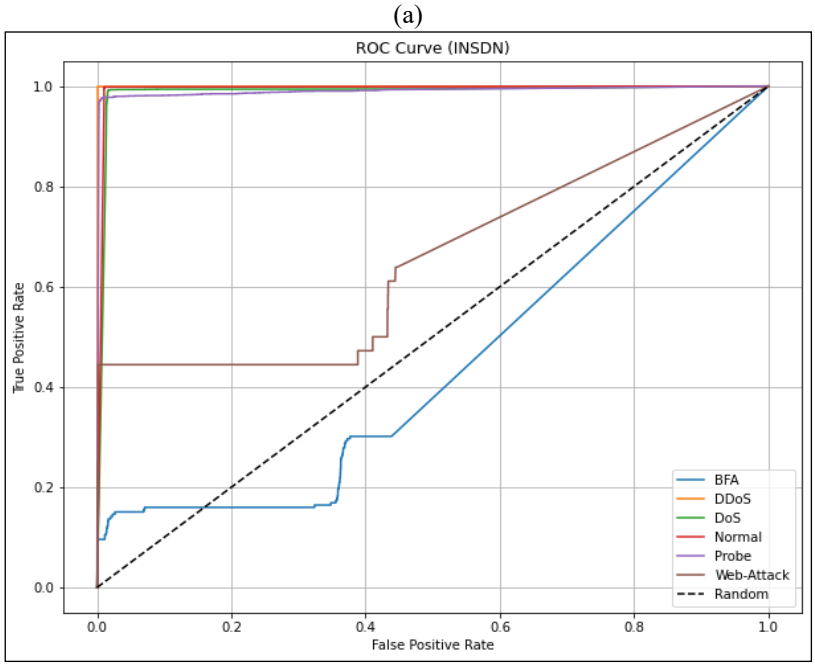
In order to compare the proposed TE2T with existing work, a performance evaluation has been made between the proposed TE2T and the recent deep learning based intrusion detection frameworks for SDN and hybrid network environments. The papers provide an overview of certain modern strategies such as transformer architectures, GAN-supported data balancing process, BiLSTM networks with attentive swelling. Using the widely-used benchmark datasets CICIDS2017, CIC-DDoS-2019, NSL-KDD, and INSDN, the comparison emphasizes the detection performance, the robustness against class imbalance, and the multi-class applicability. Table 2 summarizes the comparison of TE2T with these representative methods.

Table 1. The proposed Trust-Enhanced TabTransformer (TE2T) model's performance across datasets

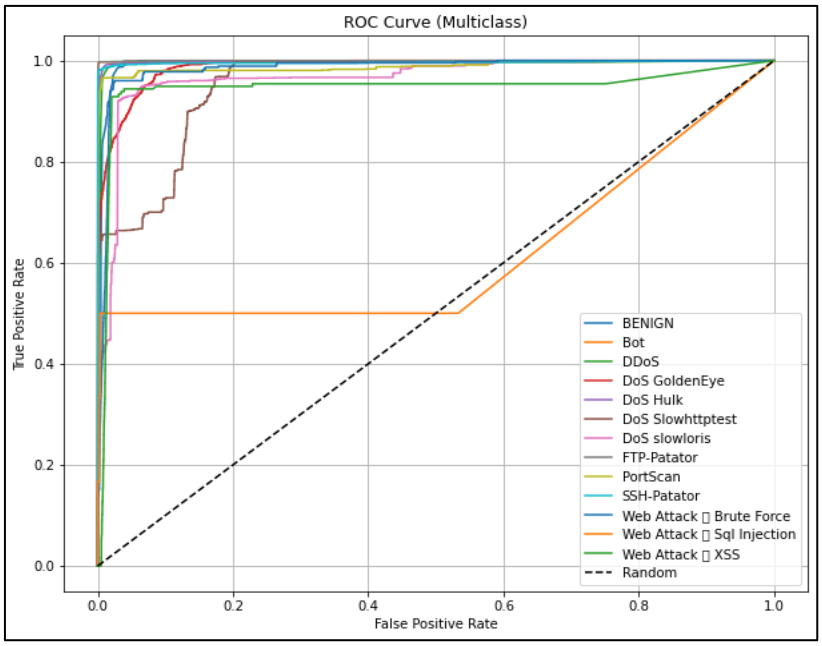
Dataset	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	ROC-AUC (%)
CICIDS2017	97.98	97.86	97.98	97.80	99.42
INSDN	97.91	96.94	97.91	97.42	98.54
UNSW-NB15	99.27	99.20	99.27	99.22	82.46



(a)



(b)



(c)

Figure 4. Proposed Trust-Enhanced TabTransformer (TE2T) model ROC curves on (a)INSDN (b)UNSW-NB15 (c)CICIDS2017 datasets

Table 2. Performance comparison with state-of-the-art Software-Defined Networking (SDN) intrusion detection system (IDS) methods

Ref	Dataset	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	ROC-AUC (%)
[15]	CICIDS2017	95.66	92.44	78.47	83.18	-
	CIC-DDOS-2019	96.48	95.70	86.73	90.05	
[16]	NSL-KDD	95.85	96.43	95.78	96.87	-
[17]	INSDN	86.00	-	-	-	-
TE2T (Proposed)	CICIDS2017	97.98	97.86	97.98	97.80	99.42
	INSDN	97.91	96.94	97.91	97.42	98.54
	UNSW-NB15	99.27	99.20	99.27	99.22	82.46

Note: Following common practice in the literature, baseline results are reported from original studies when unified experimental settings are not available

Table 2 presents a quantitative comparison between the proposed TE2T framework and recent SDN intrusion detection approaches. Unlike existing methods, TE2T consistently reports comprehensive multi-class metrics, including ROC-AUC, across heterogeneous datasets, demonstrating superior robustness and generalization.

7. DISCUSSION

Experimental results show that the proposed TE2T consistently performs well on intrusion detection under heterogeneous network environments. TE2T leverages this feature by incorporating a trust-aware mechanism directly in the attention process to boost reliable traffic patterns and suppress noisy or fence flows, which is especially suited for SDN scenarios since traffic may come from various and untrustworthy sources.

Therefore, the high accuracy, weighted F1-scores, and precision values shown in Table 1 across CICIDS2017, INSDN and UNSW-NB15, confirm the strength of the model multi-class intrusion detection, with a low false-positive output. In addition, the ROC-AUC curves shown in Figure 4 demonstrate significant degree of class separability in both imbalanced and complex attack environment. The results also reflect the advantage of our Trust Score-based attention, adding a reliability-aware diversity dimension on top of standard feature representation learning typically adopted in existing CNN-, RNN-, and attention-based IDS frameworks.

A quantitative comparison of the proposed TE2T model against the recent works in SDN intrusion detection is provided in Table 2. In general, across heterogeneous datasets, TE2T achieves consistently strong performances, which confirms the effectiveness of integrating trust-aware attention into the basic Transformer architecture. TE2T therefore provides competitive or superior detection performance than [15], on CICIDS2017, especially in terms of ROC-AUC, while keeping a simpler architecture than Transformer-based models which rely upon complex hybrid structures, or GAN-assisted data balancing.

TE2T achieves strong performance on the INSDN dataset for SDN-native scenarios, outperforming recurrent and attention-based baselines which do not explicitly model traffic reliability [17]. Incorporating trust information directly into the attention mechanism is likely to improve robustness when confronted with controller-level traffic aggregation. Additionally, unlike many methods assessed solely against outdated benchmarks such as NSL-KDD [16], TE2T generalizes well to contemporary and realistic datasets, further confirming its deployment feasibility for practical SDN solutions. Despite the strong performance, the proposed approach has some limitations. In particular, performance may

be affected in scenarios with high class overlap or highly imbalanced data distributions, as observed in datasets such as UNSW-NB15. In addition, the reliance on offline datasets limits the ability to fully assess real-time behavior in dynamic SDN environments. Future work will focus on addressing these challenges by exploring adaptive thresholding mechanisms and validating the model in real-time SDN deployments.

8. CONCLUSION

This paper introduces a Trust-Enhanced TabTransformer framework, named TE2T, for intrusion detection in SDN environments. The proposed method explicitly addresses one of the major shortcomings of the traditional deep-learning based IDSs, by incorporating a Trust Score into the Transformer attention mechanism to account for the trustworthiness of network flows. The performance evaluation against other heterogeneous datasets CICIDS2017, INSDN and UNSW-NB15 via extensive experiments showed that TE2T achieves high accuracy, precision and weighted F1-scores with very highs and robust ROC-AUC characteristics.

TE2T provides a simpler and more effective reliability-aware design that is also more generalizable to SDN-native traffic compared to existing SDN intrusion detection methods based on complex hybrid architectures or rebalancing techniques. These results demonstrate that attention-based architectures must incorporate trust modeling for any real-world SDN security deployment. Further work will investigate expanding the proposed framework to federated and distributed SDN environments, integrating adaptive trust dynamics, and assessing scalability with practical controller workloads and in real-time.

REFERENCES

[1] Haji, S.H., Zeebaree, S.R., Saeed, R.H., Ameen, S.Y., Shukur, H.M., Omar, N., Yasin, H.M. (2021). Comparison of software defined networking with traditional networking. Asian Journal of Research in Computer Science, 9: 1-18. https://doi.org/10.9734/AJRCOS/2021/v9i230216

[2] Latif, Z., Sharif, K., Li, F., Karim, M.M., Biswas, S., Wang, Y. (2020). A comprehensive survey of interface protocols for software defined networks. Journal of Network and Computer Applications, 156: 102563. https://doi.org/10.1016/j.jnca.2020.102563

[3] Mahdi, S.S., Abdullah, A.A. (2022). Enhanced security of software-defined network and network slice through hybrid quantum key distribution protocol. Infocommunications Journal, 14(3): 9-15. https://doi.org/10.36244/ICJ.2022.3.2

[4] Chica, J.C.C., Imbachi, J.C., Vega, J.F.B. (2020). Security in SDN: A comprehensive survey. Journal of Network and Computer Applications, 159: 102595. https://doi.org/10.1016/j.jnca.2020.102595

[5] Al-Ameer, A., Asraa, A., Bhaya, W.S. (2023). Intelligent intrusion detection based on multi-model federated learning for software defined network. International Journal of Safety & Security Engineering, 13(6): 1135. https://doi.org/10.18280/ijsse.130617

[6] Ahmed, M.R., Islam, S., Shatabda, S., Islam, A.M., Robin, M.T.I. (2022). Intrusion detection system in software-defined networks using machine learning and deep learning techniques–A comprehensive survey. https://doi.org/10.36227/techrxiv.17153213.v1

[7] Bashaa, M.H., Bhaya, W.S., Al-aaraji, N.H.K. (2025). Integration of zero trust architecture and machine learning for improving the security of software defined networking: A review. Journal of Intelligent Information Network and Cybersecurity, 1(1): 1. https://doi.org/10.65445/3106-1192.1000

[8] Tang, T.A., Mhamdi, L., McLernon, D., Zaidi, S.A.R., Ghogho, M. (2016). Deep learning approach for network intrusion detection in software defined networking. In 2016 International Conference on Wireless Networks and Mobile Communications (WINCOM), Fez, Morocco, pp. 258-263. https://doi.org/10.1109/WINCOM.2016.7777224

[9] Fadlullah, Z.M., Tang, F., Mao, B., Kato, N., Akashi, O., Inoue, T., Mizutani, K. (2017). State-of-the-art deep learning: Evolving machine intelligence toward tomorrow's intelligent network traffic control systems. IEEE Communications Surveys & Tutorials, 19(4): 2432-2455. https://doi.org/10.1109/COMST.2017.2707140

[10] Kareem, M.I., Jasim, M.N. (2023). Machine learning-based DDoS attack detection in software-defined networking. New Trends in Information and Communications Technology Applications, 1764: 264-281. https://doi.org/10.1007/978-3-031-35442-7_14

[11] Huang, X., Khetan, A., Cvitkovic, M., Karnin, Z. (2020). Tabtransformer: Tabular data modeling using contextual embeddings. arXiv preprint arXiv:2012.06678. https://doi.org/10.48550/arXiv.2012.06678

[12] Alzahrani, A.I., Al-Rasheed, A., Ksibi, A., Ayadi, M., Asiri, M.M., Zakariah, M. (2022). Anomaly detection in fog computing architectures using custom tab transformer for internet of things. Electronics, 11(23): 4017. https://doi.org/10.3390/electronics11234017

[13] Tang, T.A., Mhamdi, L., McLernon, D., Zaidi, S.A.R., Ghogho, M., El Moussa, F. (2020). DeepIDS: Deep learning approach for intrusion detection in software defined networking. Electronics, 9(9): 1533. https://doi.org/10.3390/electronics9091533

[14] Chaganti, R., Suliman, W., Ravi, V., Dua, A. (2023). Deep learning approach for SDN-enabled intrusion detection system in IoT networks. Information, 14(1): 41. https://doi.org/10.3390/info14010041

[15] Zhang, Y., Wu, X., Dong, H. (2024). Tibs: A deep-learning model for network intrusion detection for sdn environments. In 2024 9th International Conference on Computer and Communication Systems (ICCCS), Xi'an, China, pp. 419-426. https://doi.org/10.1109/ICCCS61882.2024.10603223

[16] Agrawal, A., Bhushan, B., Sharma, H., Hameed, A.A., Jamil, A. (2025). Advancing intrusion detection in software-defined networks. In 2025 1st International Conference on Secure IoT, Assured and Trusted Computing (SATC), Dayton, OH, USA, pp. 1-6. https://doi.org/10.1109/SATC65530.2025.11136867

[17] Bose, S., Gokulraj, G., Maheswaran, N., Logeswari, G., Anitha, T., Prabhu, D. (2024). Multi-layered security framework for intrusion detection system in software defined networking environment using machine learning. In 2024 15th International Conference on Computing

Communication and Networking Technologies (ICCCNT), Kamand, India, pp. 1-7. https://doi.org/10.1109/ICCCNT61001.2024.10724112
[18] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., et al. (2017). Attention is all you need. Advances in Neural Information Processing Systems, 30. https://doi.org/10.48550/arXiv.1706.03762
[19] Latibari, B.S., Nazari, N., Chowdhury, M.A., Gubbi, K.I., Fang, C., Ghimire, S., Sasan, A. (2024). Transformers: A security perspective. IEEE Access, 12: 181071-181105. https://doi.org/10.1109/ACCESS.2024.3509372
[20] Wang, X., Qiao, Y., Xiong, J., Zhao, Z., Zhang, N., Feng, M., Jiang, C. (2024). Advanced network intrusion detection with tabtransformer. Journal of Theory and Practice of Engineering Science, 4(3): 191-198. https://doi.org/10.53469/jtpes.2024.04(03).18
[21] Moustafa, N., Slay, J. (2015). UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In 2015 Military Communications and Information Systems Conference (MilCIS), Canberra, ACT, Australia, pp. 1-6. https://doi.org/10.1109/MilCIS.2015.7348942
[22] Sharafaldin, I., Lashkari, A.H., Ghorbani, A.A. (2018). Toward generating a new intrusion detection dataset and intrusion traffic characterization. ICISSp, 1(2018): 108-116. https://doi.org/10.5220/0006639801080116
[23] Elsayed, M.S., Le-Khac, N.A., Jurcut, A.D. (2020). InSDN: A novel SDN intrusion dataset. IEEE Access, 8: 165263-165284. https://doi.org/10.1109/ACCESS.2020.3022633
[24] Kingma, D.P., Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980. https://doi.org/10.48550/arXiv.1412.6980