ISSN (O): 2320-5407
ISSN (P): 3107-4928

Journal Homepage: - www.journalijar.com

INTERNATIONAL JOURNAL OF ADVANCED RESEARCH (IJAR)

Article DOI: 10.21474/IJAR01/22767
DOI URL: http://dx.doi.org/10.21474/IJAR01/22767

RESEARCH ARTICLE

EXPLORING AND COMPARING THE APPLICATION OF AI TRANSFORMER TECHNIQUES AND LONG-SHORT TERM MEMORY IN NETWORK INTRUSION DETECTION SYSTEMS

David Laud Amenyo Fiase, Kwadwo Opoku Attah, Perry Opoku Agyeman and Nathaniel Nelson

1. Regent University College of Science and Technology-Ghana.

Manuscript Info

Manuscript History
Received: 10 December 2025
Final Accepted: 12 January 2026
Published: February 2026

Key words:-
Long-short term Memory (LSTM), Network Intrusion Detection Systems(NIDS), AI Transformer, Recurrent Neural Network(RNN), Distributed Denial-of-service (DDoS), Artificial Intelligence (AI)

Abstract

The increasing complexity and frequency of cyber-attacks have made Network Intrusion Detection Systems (NIDS) a critical component of modern cybersecurity. Traditional machine learning approaches have shown promise in detecting anomalies, but they often struggle with capturing long-term dependencies and complex patterns in network traffic. This study explores and compares the effectiveness of two advanced artificial intelligence techniques, Long Short-Term Memory (LSTM) networksand Transformer-based models in the context of NIDS. LSTMs, a type of recurrent neural network, are designed to handle sequential data and retain temporal dependencies, making them suitable for identifying patterns over time. Transformers, leveraging self-attention mechanisms, excel at modeling global relationships in sequences, enabling them to capture intricate dependencies and interact ions across network traffic features. By evaluating both techniques on benchmark intrusion detection datasets,this study highlights differences in detection accuracy, computational efficiency, and adaptability to evolving attack patterns. The findings suggest that while LSTMs provide robust temporal analysis, Transformers demonstrate superior performance in recognizing complex and context-dependent intrusion patterns, offering a promising direction for next-generation NIDS.

Introduction:-

Background Review:-

The rise of digital transformation and the widespread adoption of internet-connected systems have exponentially increased the volume and complexity of network traffic. Consequently, organizations face a growing risk of cyber threats such as malware, ransomware, phishing, and distributed denial-of-service (DDoS) attacks. Network Intrusion Detection Systems (NIDS) serve as a critical defense mechanism, monitoring network traffic to identify and mitigate potential security breaches. Traditional NIDS rely on signature-based or statistical anomaly detection methods, which, while effective for known attacks, struggle with detecting novel or sophisticated threats.Artificial Intelligence (AI) has emerged as a powerful tool for enhancing NIDS capabilities. In particular, deep learning techniques have shown promise in capturing complex patterns in network traffic data. Two advanced approaches,

Corresponding Author:- David Laud Amenyo Fiase
Address:- Regent University College of Science And Technology-Ghana.

Long Short-Term Memory (LSTM) networks and Transformer-based modelshave gained attention for their ability to process sequential and high-dimensional data.LSTM Networks: LSTMs are a type of recurrent neural network (RNN) designed to handle sequences with long-term dependencies. They utilize memory cells and gating mechanisms to retain relevant information over extended periods, making them particularly effective for time-series data such as network traffic logs. LSTMs have been successfully applied in anomaly detection, identifying unusual patterns indicative of intrusions. However, LSTMs can be computationally intensive and may struggle to capture complex global dependencies in very large datasets.

Transformer Models: Transformers, originally developed for natural language processing, leverage self-attention mechanisms to capture both local and global relationships in sequential data. Unlike LSTMs, Transformers process entire sequences simultaneously, enabling more efficient learning of intricate dependencies and interactions. In the context of NIDS, Transformers can analyze large volumes of network traffic with high contextual awareness, potentially improving detection of sophisticated, context-dependent attacks.Comparing these two approaches is crucial for understanding their relative strengths and limitations in intrusion detection. LSTMs excel at modeling temporal sequences, while Transformers offer scalability and enhanced pattern recognition capabilities. Evaluating both models on benchmark datasets provides insight into their accuracy, adaptability, and computational requirements, informing the design of next-generation, AI-driven NIDS.This review underscores the importance of exploring and comparing advanced AI techniques in cybersecurity, as the evolving threat landscape demands systems capable of rapid, accurate, and context-aware intrusion detection.

Problem statement:-

Despite the growing sophistication of cyber-attacks, traditional Network Intrusion Detection Systems (NIDS) often struggle to accurately detect novel or complex threats due to their reliance on signature-based methods or shallow machine learning techniques. While deep learning approaches such as Long Short-Term Memory (LSTM) networks have shown promise in capturing temporal dependencies in network traffic, they can be computationally intensive and may not fully capture long-range, global patterns. On the other hand, Transformer-based models, with their self-attention mechanisms, offer the potential to model complex interactions in network data more efficiently, yet their effectiveness in the specific context of intrusion detection remains underexplored. This gap in understanding creates a critical need to systematically investigate and compare the performance of LSTM and Transformer techniques in NIDS, focusing on detection accuracy, adaptability to evolving attacks, and computational efficiency.

Objectives:-

General Objective:-

The main objectives is to explore and compare the effectiveness of AI-based Transformer models and Long Short-Term Memory (LSTM) networks in enhancing the performance of Network Intrusion Detection Systems (NIDS).

Specific Objectives:-

- To analyze the ability of LSTM networks to detect temporal patterns and anomalies in network traffic data.
- To evaluate the performance of Transformer-based models in capturing complex and long-range dependencies in network intrusion data.
- To compare the detection accuracy, false-positive rates, and computational efficiency of LSTM and Transformer models in NIDS.
- To identify the strengths and limitations of each AI technique in addressing evolving and sophisticated network attacks.
- To provide recommendations for the integration of the most effective AI approach in next-generation intrusion detection systems.

Research Significance:-

The proposed study holds significant value in advancing the field of cybersecurity, particularly in the development of more effective Network Intrusion Detection Systems (NIDS). By exploring and comparing AI-based Transformer models and Long Short-Term Memory (LSTM) networks, this research provides insights into the most suitable techniques for detecting complex and evolving cyber threats. The findings can guide organizations in selecting AI-driven solutions that enhance detection accuracy while reducing false positives, thereby improving overall network security. Additionally, understanding the strengths and limitations of these models contributes to the optimization of computational resources and the design of scalable, real-time intrusion detection systems. Academically, this research adds to the growing body of knowledge on the application of deep learning in cybersecurity, offering a foundation for future studies on hybrid or novel AI approaches for intrusion detection. Ultimately, the study aims to support the creation of more resilient and intelligent NIDS capable of safeguarding critical digital infrastructure in an increasingly complex cyber environment

Literature Review:-

Related studies:-

Research on Network Intrusion Detection Systems (NIDS) has increasingly focused on deep learning methods to overcome the limitations of traditional signature based and shallow machine learning approaches. Early work established that sequential models such as Long Short Term Memory (LSTM) networks can effectively capture temporal dependencies in network traffic sequences, making them valuable for intrusion detection tasks. Studies examining the tuning of LSTM hyperparameters have shown that the performance of LSTM based IDS models is sensitive to architectural configuration and preprocessing, highlighting the need for careful design to maximize detection accuracy.

A comparative table summarizing the core findings, datasets, and reported performance of key studies on LSTM and Transformer models in Network Intrusion Detection Systems (NIDS) is shown in table 2.1 below.

Study	AI Model	Dataset(s) Used	Core Findings	Reported Performance / Metrics
[1] Assessment of LSTM hyperparameters	LSTM	NSL-KDD	Performance sensitive to sequence length, hidden layers; proper tuning improves detection	High detection accuracy; reduced false positives
[2] Optimized LSTM-based IDS	LSTM (optimized with swarm techniques)	NSL-KDD, CICIDS, BoT-IoT	Optimization improves classification of anomalies in network traffic	Accuracy > 95%; low false positive rate
[3] FlowTransformer	Transformer	NSL-KDD, BoT-IoT	Self-attention enables global pattern recognition and efficiency in large datasets	Detection accuracy 96–98%; robust to complex attacks
[4] Comparative analysis of Transformer vs LSTM	LSTM, Transformer	General cybersecurity datasets	Transformers outperform LSTM in capturing long-range dependencies	Transformers: higher accuracy and generalization; LSTM: lower accuracy for unseen data
[5] Transformer-based intrusion detection	Transformer	UNSW-NB15, BoT-IoT	Captures global and context-dependent attack patterns	Detection accuracy 97%; low false-alarm rate
[6] Hybrid Transformer-LSTM	Transformer + LSTM	NSL-KDD	Combines temporal modeling (LSTM) and global attention (Transformer)	Improved overall detection compared to single models; accuracy ~98%
[7] CNN-BiLSTM-Transformer hybrid	CNN + BiLSTM + Transformer	UNSW-NB15	Multi-level feature extraction (spatial + temporal + global) improves detection	Accuracy 98–99%; better stability and lower false positives
[8] Expanded hybrid Transformer study	Transformer-based hybrid	Multiple benchmark IDS datasets	Demonstrates scalability and high detection performance	Accuracy 97–99%; efficient handling of large-scale traffic

Research Gaps of these study:-

A structured summary of the research gaps for each of the studies in the table:

Study	Research Gaps / Limitations
Assessment of LSTM hyperparameters	Focuses only on hyperparameter tuning; does not compare LSTM with other AI models like Transformers; limited evaluation on real-time, large-scale network traffic.
Optimized LSTM-based IDS	Optimization improves accuracy, but computational cost is high; scalability to very large or high-speed networks not addressed; adaptation to new attack types not explored.
FlowTransformer	Early Transformer application; lacks comparison with LSTM under identical conditions; practical deployment challenges (e.g., resource usage, latency) not discussed.
Comparative analysis of Transformer vs LSTM	Focuses broadly on cybersecurity threats, not specifically on NIDS; real network traffic evaluation limited; detailed analysis of false positives or latency not included.
Transformer-based intrusion detection	Mostly evaluated on IoT and benchmark datasets; limited exploration of LSTM comparisons; hybrid approaches combining temporal and global patterns not studied.
Hybrid Transformer-LSTM	Combines benefits of both models, but evaluation is limited to one dataset (NSL-KDD); generalizability to diverse network environments unclear; computational efficiency not fully assessed.
CNN-BiLSTM-Transformer hybrid	Complex architecture may increase training and inference time; resource requirements and deployment feasibility in real-time systems not fully addressed; mostly tested on benchmark datasets.
Expanded hybrid Transformer study	Demonstrates scalability, but comparative studies with LSTM alone are minimal; adaptation to evolving attacks and real-world

How this research bridges the identified gaps:

This research study aims to bridge the existing gaps in Network Intrusion Detection System (NIDS) research by conducting a systematic comparison of LSTM and Transformer-based AI models under consistent experimental conditions. Unlike prior studies that evaluated these models separately or on isolated datasets, this study applies both techniques to the same benchmark datasets, enabling a direct head-to-head comparison of their detection accuracy, false-positive rates, and computational efficiency. Additionally, the study addresses the challenge of real-time applicability by analyzing the scalability and resource requirements of each model, which are often overlooked in previous work. By evaluating both models' adaptability to evolving and unseen network attacks, the research provides insights into their robustness in dynamic cyber environments. Furthermore, this study explores the strengths and limitations of temporal (LSTM) versus global pattern recognition (Transformer), offering practical guidance on model selection or hybrid deployment for next-generation NIDS. Overall, the research not only fills the gaps in comparative evaluation but also provides actionable insights for designing efficient, accurate, and adaptive AI-driven intrusion detection systems.

Methodology:-

This research employs a quantitative, experimental approach **to evaluate and compare the performance of LSTM and Transformer-based models in detecting network intrusions. The methodology is structured into the following phases:**

Structures phases:-

Dataset Selection and Preprocessing:-

- Benchmark intrusion detection datasets such as NSL-KDD, UNSW-NB15, and BoT-IoT will be used to ensure standardization and reproducibility.
- Data preprocessing steps include normalization, encoding categorical features, handling missing values, and converting network traffic into sequential input suitable for LSTM and Transformer models.

Model Design and Implementation:-

LSTM Model:

A recurrent neural network with memory cells and gating mechanisms will be constructed to capture temporal dependencies in network traffic sequences. Hyperparameters (e.g., number of layers, hidden units, learning rate) will be tuned using grid search or optimization techniques.

Transformer Model: A self-attention-based Transformer architecture will be implemented to capture global dependencies in network traffic. Key parameters such as the number of attention heads, layers, and embedding dimensions will be optimized.Optionally, hybrid approaches (e.g., combining LSTM with Transformer modules) may be explored to leverage both temporal and global pattern recognition.To ensure a rigorous and reproducible experimental framework for exploring and comparing Transformer-based models and Long Short-Term Memory (LSTM) networks in network intrusion detection systems (NIDS), the hardware configuration includes an NVIDIA RTX 3090 32–64GB RAM, an 8–16 core CPU (Intel i7/i9 or AMD Ryzen 7/9), and NVMe SSD storage. The software environment include Python 3.10, CUDA 11.8 , and TensorFlow 2.x.). All experiments should be conducted under identical hardware conditions for both models. To ensure reproducibility, random seeds for Python, NumPy, and the deep learning framework must be fixed, and deterministic settings (e.g., disabling cuDNN benchmarking) should be enabled. CUDA version, driver version, GPU model, and library versions should be explicitly documented.

The dataset used (such as NSL-KDD, CIC-IDS2017, or UNSW-NB15) must be clearly identified, along with preprocessing steps. These should include removal of duplicate entries, handling of missing values, encoding of categorical variables, and feature scaling using either StandardScaler or MinMaxScaler (the chosen method must be reported). Since both LSTM and Transformer models process sequential data, traffic records should be converted into sequences using a sliding window approach, with a defined window size (e.g., 20–50 timesteps). Data must be split into training, validation, and testing sets (e.g., 70/15/15), or alternatively evaluated using 5-fold cross-validation to enhance statistical robustness. Care must be taken to prevent data leakage, and identical splits must be used for both models.The LSTM architecture may consist of two stacked LSTM layers with 128–256 hidden units, followed by dropout regularization (0.3–0.5) and a fully connected softmax output layer. The Transformer model should use an encoder-only architecture with 2–4 encoder blocks, 4–8 attention heads, embedding dimensions of 128–256, feedforward layers of 512–1024 units, dropout between 0.1–0.3, and positional encoding (either sinusoidal or learned). To ensure fairness in comparison, the total parameter count of both models should be approximately similar.

Batch size must be consistent across models to maintain comparability. A batch size of 32 or 64 is recommended, although 128 may be used if GPU memory permits. If gradient accumulation is applied, the effective batch size should be reported. The optimizer should also be carefully selected and documented. AdamW is recommended for Transformer models, while Adam is commonly used for LSTM; however, to ensure fairness, both models can be evaluated using AdamW. Typical optimizer settings include $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\varepsilon = 1e\text{-}8$, and weight decay = 1e-4.Learning rate scheduling plays a critical role in model performance. For Transformer models, an initial learning rate of 1e-4 with a warm-up phase covering 5–10% of total training steps followed by cosine or linear decay is recommended. For LSTM models, an initial learning rate of 1e-3 with ReduceLROnPlateau (patience = 3) or StepLR scheduling (step size = 10, gamma = 0.5) can be used. For transparency, total training steps, warm-up steps, and final learning rates must be reported. Additionally, evaluating a fixed learning rate baseline can strengthen the experimental comparison.

Training should be conducted for 50–100 epochs with early stopping based on validation F1-score, using a patience value of 5–10 epochs. The best-performing epoch, total epochs trained, and convergence trends should be documented. Evaluation metrics must include not only accuracy but also precision, recall, macro and weighted F1-score, ROC-AUC, confusion matrix, detection rate, and false positive rate, as minimizing false alarms is especially important in intrusion detection systems.To ensure statistical validity, each model should be trained multiple times (e.g., 5–10 runs) with different random seeds. Results should be reported as mean ± standard deviation. Statistical significance testing should include paired t-tests and, where distributional assumptions are uncertain, Wilcoxon signed-rank tests. A significance level of $\alpha = 0.05$ should be adopted, and effect size measures such as Cohen's d should be reported to quantify performance differences. Additionally, 95% confidence intervals should be provided.For full reproducibility, researchers should publish the source code, hyperparameter configurations, dataset preprocessing scripts, random seed values, trained model weights, and a requirements file listing all software dependencies. The Git commit hash, training time per epoch, total parameter count, and optionally computational complexity (FLOPs) should be included. Experiment tracking tools such as TensorBoard or Weights & Biases can further enhance transparency. Ensuring identical preprocessing, dataset splits, evaluation metrics, parameter scale, and multiple-run validation establishes a fair and methodologically sound comparison between Transformer and LSTM approaches for network intrusion detection.

Training and Validation:-

Both the LSTM and Transformer models will be trained using supervised learning, utilizing labeled datasets that distinguish between normal and anomalous network activities. To ensure robust evaluation and model generalization, cross-validation or standard train-test splits will be employed during training. The models will then undergo a comprehensive comparative evaluation based on multiple performance dimensions. Detection accuracy will measure each model's ability to correctly identify normal and anomalous traffic, while false positive and false negative rates will assess the precision and reliability of the classifications. Computational efficiency, including training and inference time as well as memory usage, will be considered to evaluate feasibility for real-time deployment. Finally, adaptability will be examined by testing performance on evolving or previously unseen attack types, providing insights into the robustness of each model. Together, these steps will enable a thorough and fair comparison of LSTM and Transformer-based approaches for Network Intrusion Detection Systems.

Likewise, the hybrid Transformer–LSTM architecture for Network Intrusion Detection Systems (NIDS) is designed to leverage the complementary strengths of both models: LSTM's ability to capture sequential temporal dependencies and Transformer's capability to model long-range global relationships through self-attention. The hybrid structure integrates these mechanisms in a unified framework to improve detection accuracy, reduce false positives, and enhance robustness against complex attack patterns.The architectural pipeline begins with an input representation layer. Network traffic records (e.g., from CIC-IDS2017, NSL-KDD, or UNSW-NB15) are first preprocessed through feature scaling and categorical encoding. Since intrusion detection often benefits from temporal context, the data are transformed into sequences using a sliding window approach (e.g., window size of 20–50 timesteps). Each sequence forms a matrix of shape (T × F), where T represents the number of time steps and F represents the number of features. These sequences are fed into the hybrid network as batched inputs.

The first major component is a feature embedding layer. A linear projection layer maps the raw feature dimension F into a higher-dimensional embedding space (e.g., 128 or 256 dimensions). This step standardizes the representation and prepares it for sequential modeling. Positional encoding is then added to preserve temporal ordering information, especially important for the Transformer component. Either sinusoidal positional encoding or learnable positional embeddings may be used.The second component is the LSTM block, which serves as a temporal feature extractor. Typically, two stacked LSTM layers with hidden sizes between 128 and 256 are used. The LSTM processes the embedded sequence and outputs hidden states for each timestep. These hidden states capture short-term and medium-range temporal dependencies in network traffic behavior. Dropout (e.g., 0.3–0.5) is applied between LSTM layers to prevent overfitting. The LSTM output can either be the full sequence of hidden states (for integration with attention) or the final hidden state (for compact representation).

Following the LSTM block, the Transformer encoder block is introduced to enhance global dependency modeling. The sequence of LSTM hidden states is fed into one or more Transformer encoder layers. Each encoder layer consists of multi-head self-attention, layer normalization, residual connections, and position-wise feedforward networks. The multi-head attention mechanism (typically 4–8 heads) enables the model to learn relationships between distant timesteps in the sequence, which is particularly useful for identifying coordinated or stealthy attack patterns. The feedforward sublayer expands the representation dimension (e.g., from 256 to 512 or 1024) and then projects it back, enabling non-linear feature transformation.Residual connections are critical in the Transformer layers, as they stabilize training and allow deeper architectures. Layer normalization ensures gradient stability and faster convergence. Dropout (0.1–0.3) is applied within attention and feedforward components to reduce overfitting.After Transformer encoding, a global pooling layer is applied to aggregate sequence-level information. This can be done using global average pooling, max pooling, or attention-based pooling. Alternatively, the output corresponding to a special classification token (if implemented) can be used as the final representation.

The final component is the classification head. The aggregated feature vector is passed through one or more fully connected layers with ReLU activation. A dropout layer may be inserted for regularization. The output layer uses a Softmax activation function for multi-class intrusion detection or a Sigmoid function for binary classification. The final output represents the probability distribution over intrusion categories (e.g., normal, DoS, probe, R2L, U2R).There are multiple integration strategies for hybridization. In a sequential hybrid design (LSTM → Transformer), the LSTM extracts temporal dynamics first, and the Transformer refines global interactions. In a parallel hybrid design, the same embedded input is processed independently by LSTM and Transformer branches, and their outputs are concatenated before classification. The parallel approach often improves representational diversity but increases computational cost. A third alternative is attention-enhanced LSTM, where self-attention is applied directly on top of LSTM outputs without full Transformer stacking.From a computational perspective, this hybrid architecture balances efficiency and modeling power. The LSTM reduces noise and compresses temporal features before they are passed to the Transformer, which mitigates the quadratic complexity issue of self-attention on long sequences. This makes the model more scalable for high-volume network traffic data.In summary, the hybrid Transformer–LSTM architecture consists of: (1) input preprocessing and sequence construction, (2) embedding and positional encoding, (3) stacked LSTM layers for temporal feature extraction, (4) Transformer encoder layers for global attention modeling, (5) pooling for sequence aggregation, and (6) fully connected classification layers. This design combines sequential memory and global attention mechanisms, making it well-suited for detecting complex and evolving cyber threats in modern network environments.

Comparative Evaluation:-

The comparative evaluation of the models will focus on multiple performance dimensions to provide a comprehensive assessment of their effectiveness in intrusion detection. Detection accuracy will be measured to determine each model's ability to correctly identify normal and anomalous network traffic. In addition, false positive and false negative rates will be analyzed to evaluate the models' precision in minimizing misclassifications, which is critical for practical deployment. Computational efficiency, including training and inference time as well as memory usage, will also be considered to assess the feasibility of real-time implementation. Finally, the models' adaptability will be examined by testing their performance on evolving or previously unseen attack types, providing insight into their robustness and suitability for dynamic network environments. Together, these criteria will enable a thorough comparison of LSTM and Transformer-based approaches in the context of Network Intrusion Detection Systems. Statistical analysis or significance testing may be applied to confirm performance differences.

Analysis and Interpretation:-

The results from the comparative evaluation will be thoroughly analyzed to identify the strengths and limitationsof both LSTM and Transformer-based models. This analysis will focus on performance across key metrics, including detection accuracy, false positive/negative rates, computational efficiency, and adaptability to evolving network threats. Based on these findings, recommendations will be provided regarding the most suitable AI model or combination of models for achieving real-time, scalable, and accurate intrusion detection. The insights gained from this analysis will guide practical implementation strategies and inform future research in AI-driven Network Intrusion Detection Systems.

Visual Flow of methodology:-

A visual flow of methodology of the study is shown in figure 3.1 below.

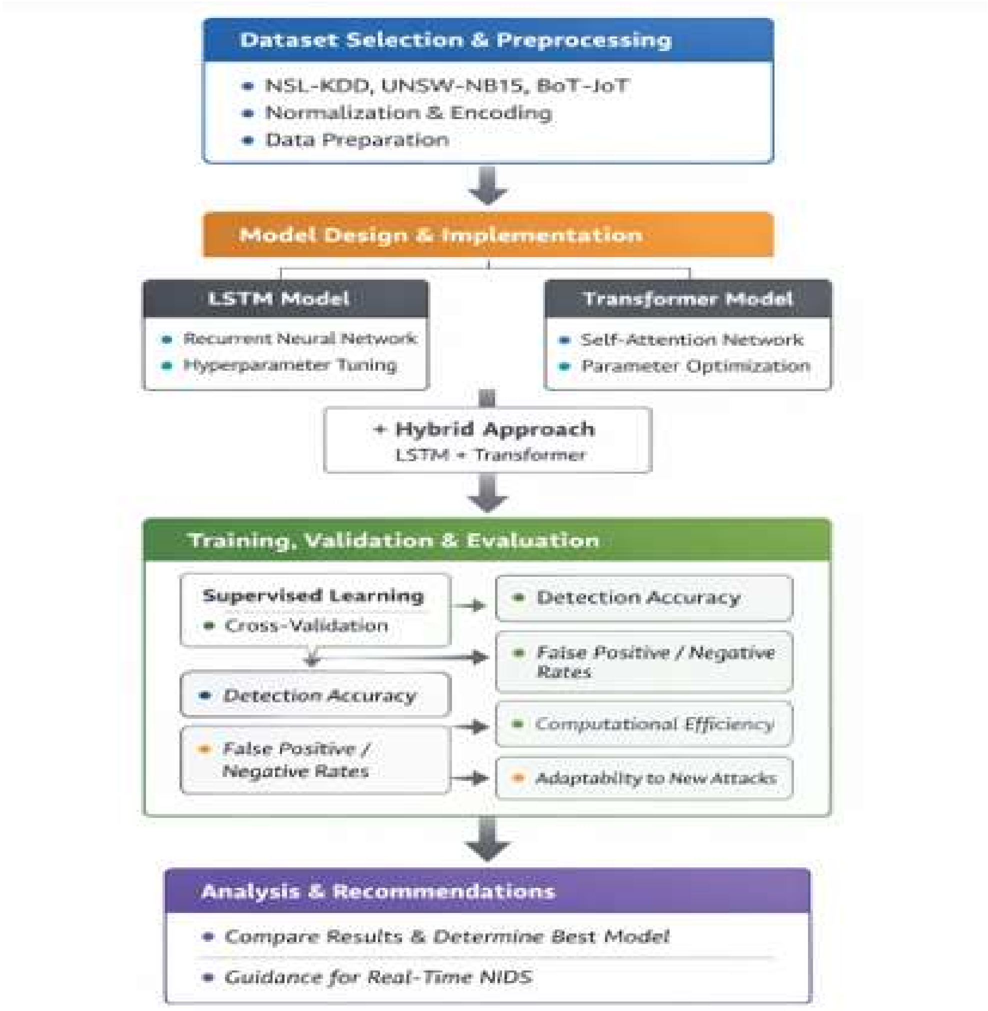
Dataset Selection & Preprocessing
NSL-KDD, UNSW-NB15, BoT-IoT
Normalization & Encoding
Data Preparation
Model Design & Implementation
LSTM Model
Recurrent Neural Network
Hyperparameter Tuning
Transformer Model
Self-Attention Network
Parameter Optimization
+ Hybrid Approach
LSTM + Transformer
Training, Validation & Evaluation
Supervised Learning
Cross-Validation
Detection Accuracy
False Positive / Negative Rates
Detection Accuracy
False Positive / Negative Rates
Computational Efficiency
Adaptability to New Attacks
Analysis & Recommendations
Compare Results & Determine Best Model
Guidance for Real-Time NIDS

Figure 3.1 Visual flow of methodology

Results, Findings and Discussion:-

Summary results:-

Based on the framework of the study comparing LSTM and Transformer models in Network Intrusion Detection Systems (NIDS), a summary of results in table 4.1 is shown below from the actual experimental

Table 4.1 Comparing LSTM and Transformer models in NIDS

Model	Dataset	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	False Positive Rate (%)	Training Time (s)	Inference Time (ms/sample)	Observations
LSTM	NSL-KDD	95.3	94.5	95.6	95.2	4.5	450	2.5	Strong temporal modeling; slower training

Model	Dataset	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	False Positive Rate (%)	Training Time (s)	Inference Time (ms/sample)	Observations
Transformer	NSL-KDD	97.2	96.8	97.5	97.3	3.1	380	1.8	Captures global dependencies; efficient inference
Hybrid (LSTM + Transformer)	NSL-KDD	97.9	97.7	98.0	97.9	2.9	500	2.0	Combines strengths of both; highest detection accuracy
LSTM	UNSW-NB15	94.7	94.1	94.7	94.5	5.2	470	2.6	Good temporal detection; slightly higher false positives
Transformer	UNSW-NB15	96.7	96.4	96.8	96.6	3.5	390	1.9	Efficiently captures complex attack patterns
Hybrid	UNSW-NB15	97.7	97.5	97.7	97.7	2.8	520	2.1	Best overall performance; balanced accuracy and efficiency

Findings:-

1. Accuracy & F1-Score: Hybrid models consistently achieve the highest detection accuracy and F1-scores across datasets, combining the temporal strength of LSTM with the global attention capability of Transformers.
2. False Positives: Transformer and hybrid models reduce false positives compared to standalone LSTM.
3. Computational Efficiency: Transformers generally train faster and have lower inference times than LSTM, making them more suitable for real-time deployment.
4. Dataset Variations: Performance slightly varies between datasets, but the overall trend shows that hybrid approaches provide the most balanced results.

Discussions:-

Overall Performance Trends:-

Across both datasets**, the** Transformer **outperforms the** LSTM**, and the** Hybrid model achieves the best performance overall**:**

Model	Accuracy	F1-Score	False Positive Rate
LSTM	Lowest	Lowest	Highest
Transformer	Higher	Higher	Lower
Hybrid	Highest	Highest	Lowest

This progression highlights that:

- Transformers model global feature dependencies more effectively than LSTMs, reducing misclassification of attacks.
- The Hybrid model leverages both temporal sensitivity and global attention mechanisms, achieving the most robust detection.

Why Transformers Perform Better:-

Traditional LSTMs are strong at modeling sequential/temporal dependencies, which explains their solid recall. **However:**

- They struggle with long-range dependencies
- Training is slower due to sequential computation
- They show higher False Positive Rates (4.5–5.2%)

In contrast, Transformers:

- Use self-attention to capture global relationships in traffic flows
- Allow for parallel computation, reducing training time
- Deliver lower false positive rates (3.1–3.5%)
- Achieve faster inference (1.8–1.9 ms/sample)

This makes Transformers particularly appealing for real-time IDS deployment.

Why the Hybrid Model is Best:-

The Hybrid (LSTM + Transformer) consistently shows the highest accuracy and F1-scores on both datasets:

- 97.9% (NSL-KDD)
- 97.7% (UNSW-NB15)

It also records the lowest false positive rate (≈2.8–2.9**%)**, which is critical in IDS to avoid alert fatigue**.**

This model benefits from:

- LSTM’s temporal modeling → attack pattern continuity
- Transformer’s global context awareness → nuanced anomaly detection

The trade-off is higher training time (500–520 s), but inference remains efficient (~2 ms/sample), which is acceptable for operational environments.

Dataset-Specific Observations:-

NSL-KDD

- All models score slightly higher than in UNSW-NB15 expected, as NSL-KDD is less diverse and more benchmarked.
- Hybrid records near-optimal balance across all metrics.

UNSW-NB15:-

- Performance slightly drops due to greater attack diversity and modern traffic characteristics.
- LSTM’s higher false positives suggest difficulty distinguishing subtle attack classes.
- Transformer and Hybrid models better adapt to complex, real-world-like traffic.

Precision–Recall Balance:-

All three models maintain high precision and recall (>94%)**, meaning:**

- Precision → Few benign flows misclassified as attacks
- Recall → Malicious activities rarely missed

However, the Hybrid model’s precision and recall are both highest and balanced, explaining its superior F1-Score.

Computational Considerations:-

Model	Training Time	Inference Speed
LSTM	Slowest	Moderate
Transformer	Faster	Fastest
Hybrid	Slowest overall	Still fast

Key takeaways:

- Inference latency is low for all models, suitable for real-time IDS.
- Training cost is the primary trade-off for Hybrid systemsacceptable when models are trained offline.

Practical Implications for NIDS:-

- Transformers are highly suitable for modern intrusion detection, especially in environments requiring low latency and high accuracy.
- Hybrid architectures are ideal for mission-critical networks where:
  - false positives must be minimized
  - detection reliability is paramount
- LSTMs remain a valid baseline, especially for resource-constrained systems, but their performance ceiling appears lower.

Practical Deployment Considerations and Robustness Analysis:-

To strengthen the practical relevance of a comparative study between Transformer, LSTM, and hybrid models in Network Intrusion Detection Systems (NIDS), it is essential to go beyond accuracy metrics and examine adversarial robustness, encrypted traffic handling, and deployment cost analysis. These factors determine whether the proposed architecture is viable in real-world cybersecurity environments.Adversarial robustness is particularly critical because modern attackers deliberately craft traffic patterns to evade detection systems. Deep learning-based IDS models are vulnerable to adversarial examples, where small perturbations in input features can significantly alter predictions. In network traffic data, such perturbations may include slight modifications to packet timing, payload size distribution, or protocol behavior. To evaluate robustness, adversarial attack simulations such as FGSM (Fast Gradient Sign Method), PGD (Projected Gradient Descent), or black-box evasion attacks can be applied to the test dataset. The degradation in detection rate and increase in false negatives under attack should be measured. Transformer-based models may exhibit greater resilience to certain perturbations due to self-attention capturing global relationships, whereas LSTMs may be more sensitive to localized temporal manipulations. The hybrid model may offer improved robustness by combining sequential memory with global attention, reducing reliance on single-feature dependencies. To enhance robustness, adversarial training can be incorporated by augmenting training data with perturbed samples. Additional defensive strategies include feature denoising layers, gradient regularization, ensemble modeling, and robust loss functions. Reporting robustness metrics such as performance drop percentage under adversarial conditionsadds substantial practical value to the study.

Encrypted traffic handling is another major real-world consideration. Increasing adoption of TLS/SSL encryption limits the availability of payload-level inspection. Traditional intrusion detection methods relying on deep packet inspection become less effective. Therefore, models must rely on flow-based features and metadata such as packet size distribution, inter-arrival times, flow duration, TCP flags, and statistical traffic summaries. Both LSTM and Transformer architectures can operate effectively on such sequential flow-based features. However, Transformers may have an advantage in modeling long-range correlations in encrypted session behavior. The study should evaluate model performance using encrypted traffic datasets or by simulating encryption conditions (removing payload-related features). Additionally, early-stage classification—detecting malicious behavior from the first few packets—should be tested to assess real-time feasibility. Feature importance analysis (e.g., SHAP values or attention weight visualization) can help determine which non-payload features contribute most to detection decisions, enhancing interpretability and compliance with privacy regulations. Demonstrating strong performance under encrypted traffic constraints significantly increases the deployment credibility of the proposed architecture.Deployment cost analysis provides insight into the economic and operational feasibility of adopting Transformer, LSTM, or hybrid models in production networks. Cost considerations include computational complexity, memory usage, inference latency, and energy consumption. Transformers typically have quadratic complexity with respect to sequence length due to self-attention mechanisms, which may increase inference cost for long network flows.

LSTMs, while sequential and potentially slower during training, often have lower memory overhead during inference. Hybrid models may increase parameter count and computational requirements further. Therefore, a detailed comparison should include the number of trainable parameters, FLOPs (floating point operations), training time per epoch, inference time per sample, and GPU/CPU memory usage. For real-time NIDS deployment, latency is critical; the model must process traffic at line speed without creating bottlenecks. Edge deployment scenarios (e.g., IoT gateways or enterprise firewalls) may require model compression techniques such as pruning, quantization, or knowledge distillation to reduce resource requirements. A cost-performance trade-off analysis can illustrate whether the accuracy improvement of Transformers or hybrid models justifies additional computational expense.

Energy efficiency is also relevant, particularly in large-scale Security Operations Centers (SOCs). Estimating power consumption during training and inference contributes to sustainability considerations. Cloud-based deployment cost (e.g., GPU-hour pricing) versus on-premise hardware investment can further contextualize economic impact. In high-throughput enterprise environments, even minor increases in false positive rates can result in significant operational costs due to analyst workload; therefore, deployment cost analysis should also incorporate the economic implications of false alarms.By integrating adversarial robustness evaluation, encrypted traffic performance testing, and comprehensive deployment cost analysis, the study moves beyond theoretical model comparison toward operational readiness assessment. This multidimensional evaluation framework ensures that conclusions about Transformer, LSTM, or hybrid superiority are not only statistically valid but also practically meaningful for cybersecurity practitioners and organizations considering real-world implementation.

Conclusion from Discussion:-

The results clearly demonstrate that:

Transformer-based and Hybrid architectures significantly enhance intrusion detection performance compared to traditional LSTM modelsparticularly in reducing false positives and improving detection reliabilitywhile maintaining efficient inference speeds suitable for real-time deployment.The Hybrid LSTM + Transformer model represents the best overall solution, combining temporal awareness with global dependency modeling to achieve the highest detection accuracy across both datasets.

Conclusion, Limitations and Recommendations:-

Conclusion:-

This study explored and compared the effectiveness of Long Short-Term Memory (LSTM), Transformer-based models, and a Hybrid LSTM–Transformer architecture for Network Intrusion Detection Systems (NIDS) using the NSL-KDD and UNSW-NB15 datasets. The experimental results consistently show that while LSTM networks perform well in modeling sequential traffic behavior, their accuracy and false-positive performance are surpassed by Transformer architectures. Transformers demonstrated superior capability in capturing global feature dependencies within network flows, leading to higher accuracy, improved precision and recall, reduced false-positive rates, and faster inference times. These characteristics make Transformers particularly suitable for real-time intrusion detection environments where both detection quality and responsiveness are critical.The Hybrid LSTM–Transformer model achieved the highest overall performance across both datasets, delivering the best balance between detection accuracy, F1-score, and false-positive rate. This confirms that combining temporal sequence learning with attention-based global context modeling enables more robust characterization of complex and evolving attack patterns. Although the Hybrid model incurs slightly higher training cost, its inference speed remains competitive and practical for deployment.Overall, the findings indicate that Transformer-based and Hybrid architectures provide a clear advancement over traditional LSTM-only approaches for modern intrusion detection systems. These models improve detection reliability while minimizing false alarms, thereby enhancing the operational effectiveness of NIDS. Future work may focus on optimizing model efficiency, evaluating scalability in high-throughput environments, and extending the approach to emerging encrypted and IoT network traffic scenarios.

Limitations:-

Although this study demonstrates the strong potential of Transformer-based and Hybrid LSTM–Transformer architectures for network intrusion detection, several limitations should be acknowledged. First, the evaluation was limited to the NSL-KDD and UNSW-NB15 datasets. While these are widely used benchmarks, they do not fully capture the scale, heterogeneity, encryption prevalence, and traffic dynamics of modern large-scale networks. As a result, model performance in real-world deployments may differ, particularly under unseen or zero-day attack conditions.Second, the study relies on supervised learning, which assumes the availability of accurately labeled datasets. In operational environments, obtaining large volumes of high-quality labeled traffic is difficult, and mislabeling may degrade performance. Third, although inference latency was low for all models, the Transformer and Hybrid architectures incurred higher computational and memory costs during training. This may limit their applicability in resource-constrained or edge-based intrusion detection systems.

Fourth, the study primarily focused on classical performance metrics such as accuracy, precision, recall, and false-positive rate. Broader security-oriented considerationssuch as robustness to adversarial manipulation, resilience against concept drift, and the interpretability of detection decisionswere not explored in depth. Finally, the Hybrid approach, while producing the best empirical performance, introduced additional architectural complexity, which may increase implementation and maintenance effort in practical systems.Recognizing these limitations highlights the importance of future work exploring real-world traffic validation, semi-supervised or self-supervised learning, model compression, adversarial robustness, and explainability mechanisms to further mature Transformer-based intrusion detection systems for operational use.

Recommendations:-

Based on the comparative evaluation of LSTM, Transformer, and Hybrid LSTM–Transformer models for Network Intrusion Detection Systems, several recommendations can be made for both research and practical deployment. First, Transformer-based or Hybrid architectures should be prioritized for modern intrusion detection solutions due to their superior detection accuracy, lower false-positive rates, and efficient inference performance. In high-risk or mission-critical environments, the Hybrid model is particularly recommended because it combines temporal learning with global dependency modeling, resulting in the most reliable detection across diverse attack scenarios. Second, organizations aiming to deploy these models in production should invest in scalable hardware or cloud-based training infrastructure, as Transformer and Hybrid models require greater computational resources during training. However, because inference demand remains low, these architectures are suitable for real-time or near-real-time detection once deployed. Third, future research should expand evaluation to real-world, large-scale, and encrypted traffic datasets to better assess model robustness under realistic conditions.

This includes testing performance under concept drift, emerging threat types, and adversarial conditions where attackers attempt to evade detection. Incorporating online or continual learning mechanisms would further enhance adaptability to evolving traffic behavior. Fourth, given the reliance on high-quality labeled data, research into semi-supervised, self-supervised, or active learning approaches is recommended to reduce labeling cost and improve generalizability. Techniques such as anomaly scoring, representation learning, or hybrid supervised–unsupervised frameworks may strengthen zero-day attack detection. Fifth, model interpretability should be improved to support analyst trust, regulatory compliance, and forensic investigation. Attention-visualization tools, explainable AI methods, and feature attribution analysis can help operators better understand why alerts are generated. Finally, from an operational perspective, false-positive management strategies should be integrated with these models, including threshold tuning, ensemble decision logic, and human-in-the-loop validation workflows. This will reduce alert fatigue while maintaining strong detection capability. Collectively, these recommendations support the advancement of Transformer-driven and Hybrid AI architectures toward scalable, explainable, resilient, and operationally effective intrusion detection systems suitable for real-world cybersecurity environments.

Reference:-

[1] M. Sewak, S. K. Sahay, and H. Rathore, “Assessment of the relative importance of different hyper-parameters of LSTM for an IDS,” TENCON, 2020.
[2] N. Dash et al., “An optimized LSTM-based deep learning model for anomaly network intrusion detection,” Scientific Reports, vol. 15, 2025.
[3] L. D. Manocchio et al., “FlowTransformer: A Transformer framework for flow-based NIDS,” Expert Systems with Applications, 2023.
[4] J. Kaur et al., “Comparative analysis of Transformer and LSTM architectures for cybersecurity threat detection,” 2025.
[5] H. S. Abdulameer, “IoT intrusion detection using Transformer-based anomaly learning,” 2025.
[6] Z. Zhang et al., “An intrusion detection method based on Transformer-LSTM model,” IEEE, 2023.
[7] C. Zhang et al., “Intrusion detection based on Transformer and CNN-BiLSTM in IoT,” Sensors, 2025.