AI-ASSISTED NETWORK INTRUSION DETECTION SYSTEMS FOR REAL-TIME THREAT IDENTIFICATION

Ashok Poudel[1], Dr. Umesh Kumar[2] and Prof. Anilkumar Patel[3]
[1]Department of CSE, Faculty of Engineering & Technology Technology, Vadodara, Gujarat
[2]Department of CSE, Faculty of Engineering & Technology Technology, Vadodara, Gujarat
[3] Department of CSE, Faculty of Engineering & Technology Technology, Vadodara, Gujarat
Email : 2403032010047@paruluniversity.ac.in
anilkumar.patel2986@paruluniversity.ac.in
umeshkumar.kumar38823@paruluniversity.ac.in

ABSTRACT

The increasing magnitude and complexity of network-based cyber threat have rendered real-time intrusion detection a need of high importance to the current network security frameworks. The traditional signature-based intrusion detection systems have shortcomings when it comes to detecting a zero-day attack, as well as the dynamically changing pattern of threats in a high-speed network. This paper proposes a network intrusion detection system with the help of AI to identify threats in real-time through real network traffic information. Two models (Random Forest classifier and a Long Short-Term Memory (LSTM) neural network) were used. A dataset of 148,517 network traffic records that included normal and malicious records was used to evaluate the system. It has been established through experimentation that the former model was able to achieve a precision of 95.74, recall of 96.21, accuracy of 96.08 and an average detection latency of 15 ms whereas the LSTM model performed better with a precision of 98.11, recall of 98.67 and an accuracy of 98.42 and an average detection latency of 21 ms. The results indicate the importance of AI-supported intrusion detection systems that, in addition to the high accuracy and strength of detecting intrusions, are able to operate in real-time with deep learning demonstrating a better ability to detect a complex and changing network attacks.

Keywords: AI-assisted intrusion detection, real-time threat identification, network security, machine learning, cyber threat detection

I. INTRODUCTION

Network Intrusion Detection Systems (NIDS) are important in the protection of the present-day communication infrastructures by overseeing network traffic with regard to malicious activities. The emergence of cloud computing, Internet of Things (IoT), and distributed network architecture has made cyber threats more and more complex, stealthy, and voluminous. Contemporary intruders often use sophisticated persistent attacks, polymorphic software and coordinated attacks that use vulnerabilities in heterogeneous environments. Consequently, the conventional network defense techniques are continuously forced to handle large and high-speed network environments (Vikram et al., 2024; Akbar and Zafer, 2024). These dynamic threat environment requirements call out smart and dynamic intrusion detection services that can respond and analyse in real time.

Weaknesses of Traditional and Signature-Based IDS.

The conventional intrusion detection systems are mainly based on signature-based detection or rule-based detection that is based on predefined attacks and rules as developed by the experts. Although these systems may be effective in known attacks, they proven to be severely challenged in dealing with zero-day exploits, encrypted traffic and fast changing attack patterns. Also, signature based IDS are usually characterized by high false-positives and have to be frequently updated by humans in order to be effective. These limitations make them less applicable to real-time detection in cases when detecting a threat timely and accurately is a must (Amomo, 2022; Dhanushkodi and Thejas, 2024). The combination of deep learning and supervised models has shown a higher capability to detect the more sophisticated attack patterns

and lower the number of false alarms in the dynamically changing networks (Yellepeddi et al., 2024; Samha et al., 2024).

Research Gap and Motivation

Although there is an increasing amount of literature on the AI-assisted intrusion detection, most current studies are theoretical or simulation-based, with very little attention being paid to the real-time implementation and a systematic evaluation of the performance. Implementation-driven studies are missing to confirm AI-assisted IDS in the realities of operation regarding detection latency, scalability, and resource efficiency. This disconnect indicates the necessity of feasible frameworks that can integrate solid AI models with real-time testing to determine their performance in operational network settings (James, 2025; Pamarthi, 2025).

Objectives of This Study

The main task in this research is to develop and deploy a network intrusion detection system with the help of AI that would be able to identify threats in real-time. The research also intends to conduct a comparative analysis of the selected and adopted AI models based on the performance measures, such as precision, recall, accuracy, F1-score, and detection latency. By adopting this implementation-oriented strategy, the study aims to detect which model is most applicable to the actual intrusion detection process and add some practical information about the application of AI-assisted IDS in contemporary network security settings (Ali, 2024; Raghav, 2025).

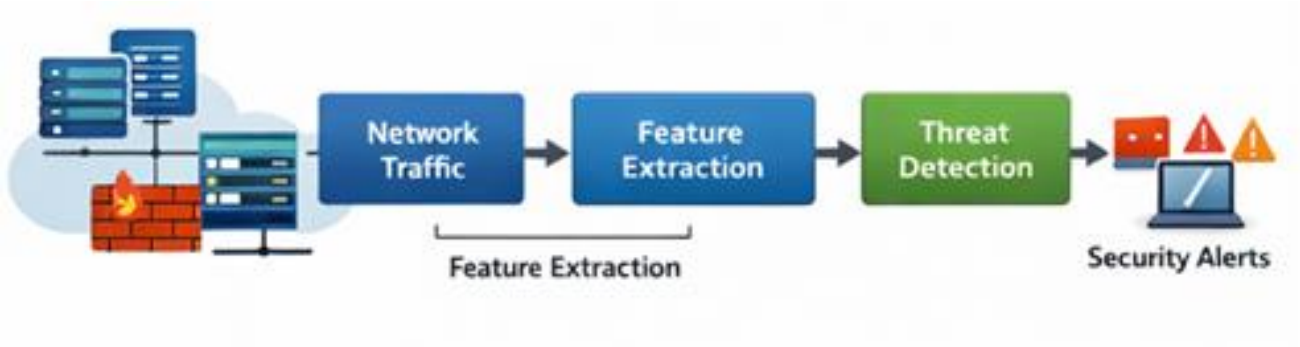

Figure 1: Conceptual Architecture of an AI-Assisted Real-Time Network Intrusion Detection System

The figure shows the high-level architecture of the AI-based network intrusion detection system, which shows the flow of network traffic, starting with data acquisition to real-time threat identification. It reflects some of the important elements including data preprocessing, feature extraction, AI-based detection models, and alert generation modules. This theoretical framework will focus on how AI can facilitate adaptive learning and real-time analysis as a framework to introduce the implementation and analysis in this research.

II. LITERATURE SURVEY

AI-Based Network Intrusion Detection Systems

It has recently been shown that there is an increasing trend of abandoning older style intrusion detection mechanisms in favor of AI-based network intrusion detection systems that can perform adaptive and intelligent threat detection. Intelligent IDS systems are founded on the application of data-driven models to identify complex patterns in network traffic and enhance the detection accuracy with time. According to the studies, AI techniques have been noted to witness significant performance gains in various aspects such as detection rate and low false positives in

terms of large-scale and dynamic network settings (Vikram et al., 2024; Yellepeddi et al., 2024). The publications underscore the usefulness of AI-based systems to overcome the shortcomings of traditional rule-based systems.

Intrusion Detection by Machine Learning Techniques.

The use of machine learning methods in intrusion detection especially in situations of anomaly-based detection, has been extensively studied. Classifying known types of attacks is usually done through supervised learning models, whereas unsupervised learning strategies are applicable in detecting an unknown or an unusual behavior. According to the previous research, machine-learning-based IDS may perform more effectively than traditional ones, when trained on relevant data, but the performance will strongly rely on the data quality and feature selection (Ali, 2024; Dhanushkodi and Thejas, 2024). Such results emphasize the need to select a model and preprocess in AI-assisted IDS.

Deep Learning and Hybrid IDS Models.

Deep learning techniques such as neural networks and hybrid networks have been brought into the limelight in that they can automatically derive high level features on raw network data. The hybrid IDS models, using both deep learning and traditional machine learning, or composite evaluation metrics, have demonstrated superior detection and strength. Studies also show that these models are able to well model non-linear network traffic relationships, resulting in enhanced performance against sophisticated attack patterns (Samha et al., 2024; Akbar and Zafer, 2024). Nonetheless, greater computational complexity is also a major issue of real-time deployment.

Cloud, IoT, and Critical Infrastructure IDS.

The use of AI-based IDS has been extended to cloud computing, IoT ecosystems, smart grid, health care systems, and critical infrastructure based on SCADA. The environments present some special security issues such as large volumes of data, diverse devices, and very stringent latency. The existing literature shows that the application of AI-assisted IDS can strengthen the security in the mentioned areas due to the ability to monitor continuously and identify threats dynamically (Al-Otaibi et al., 2025; Narsing & Srinivasan, 2025). However, domain specific constraints usually demand tailored model structures and light implementations.

Operational Problems and Real-Time Detection.

Real time intrusion detection poses great challenges on operation with regards to latency of detection, scale and ability to deploy the intrusion detector. Multiple studies note that even though AI models perform well in offline studies with high detection rates, the real-time performance of such models can be harsh due to computational and resource bottlenecks. The problem of delayed response, model scalability, and integration with the already existing network infrastructure are still troublesome issues concerning their practical implementation (Mehmood and Tariq, 2025; Olaoye, 2025). These issues specify the necessity of implementation-conscious testing of AI-assisted IDS.

Identified Research Gaps

Despite a significant body of research on AI-assisted intrusion detection, it is observable that a gap in the research on the topic is the lack of implementation-based, real-time assessment based on consistent datasets, tools, and measures. Numerous works are focused on enhancing accuracy

but do not take into consideration feasibility of deployment and real-time limitations. It is this gap that implies that systematic implementation-oriented research, which measures the efficiency of detection and the operational performance in realistic network environments is needed (Pamarthi, 2025; Raghav, 2025).

III. Methodology

General System Implementation

This paper deploys an end-to-end network intrusion detection system that is AI assisted to help in real-time threat detection. The system architecture is a pipeline that has a sequence of steps consisting of network traffic acquisition, data preprocessing, feature extraction, model training, and real-time inference. The packets that arrive to the network are constantly examined, converted to structured representations of features, and fed into trained artificial intelligence networks to distinguish between normal and malicious network traffic. The implementation focuses on low-latency processing but high detection accuracy is ensured.

Dataset and Pre-processing

The data information in this research is 148,517 network traffic records with labels gathered through a network environment monitored, which includes benign traffic and various types of malicious traffic. To have a solid model test, the dataset was split into training and testing input sets in an 80:20 split.

A series of data preprocessing steps were made to enhance the quality of data and the quality of the model. The data cleaning was also performed to remove incomplete and duplicate records.

One-hot encoding was employed to encode categorical features, e.g. protocol type and service, and minmax scaling was employed to make all features have equal ranges. The correlation analysis was used to select features and the feature importance scores based on tree-based models built the reduced feature set that retained the discriminative ability but avoided the excessive computation costs.

Implemented Models

Random Forest classifier has been chosen as one of the examples of machine learning models because it is powerful and easy to interpret. Also, to promote temporal correlations in network traffic patterns, a Long Short-Term Memory (LSTM) neural network was introduced. The two models have been selected in order to make a comparative analysis between the ordinary machine learning and deep learning methods with real-time restrictions.

Tools and Environment of implementation.

The Python version 3.10 was used to implement the system. Random Forest model was created with the help of scikit-learn library (version 1.3), whereas LSTM model was created with the help of TensorFlow (version 2.13) with the Keras API. Preprocessing and feature engineering were done using NumPy (version 1.24) and Pandas (version 2.0). The choice of these tools was based on the fact that they are stable and have a large community, as well as have been shown to work in large-scale machine learning or deep learning applications.

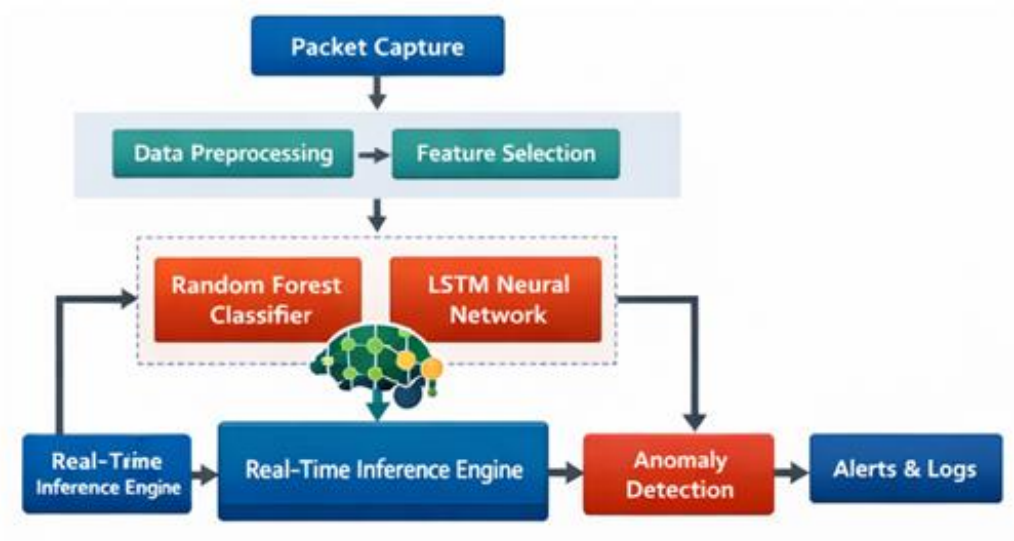

Figure 2: Implemented AI-Assisted IDS Architecture

The figure shows the applied architecture of the AI-assisted intrusion detection system, where the network traffic flow is captured, and it flows through preprocessing, feature extraction and AI-based classification. It demonstrates the way in which both Random Forest and LSTM models are incorporated into the inference layer of the real-time layer allowing to constantly monitor traffic and generate alerts. The architecture also emphasises the trade-off between computational and detection accuracy that was realised in this study.

Model Training and Online Inference.

The training subset of the dataset was used to perform model training with stratified sampling in order to maintain the balance of classes. The random Forest model was set with 200 decision trees, maximum depth of a tree of 20 and Gini impurity as the split criterion. The LSTM model had two stacked LSTM layers with a 64 and 32 units and a dense output with a sigmoid activation. Adam optimizer was used to train the LSTM at a learning rate of 0.001 and binary

cross-entropy loss. In order to avoid overfitting, the early stopping technique was used, and the testing subset was used to validate the models.

Running the trained models in a streaming detection module, which processes an incoming network traffic in fixed time windows provided real-time inference. On-the-fly predictions were made and notifications were emitted soon after some malicious activity was detected. The system has been optimized to ensure that inference latency is minimized but at the same time, the detection accuracy is preserved.

Evaluation Metrics

To measure the effectiveness of detection, accuracy, precision, and F1-score were used to estimate the performance of the system. Detection latency was recorded as a time average of classifying an instance of network traffic in real-time. The false alarm rate was computed to determine the capacity of the system to reduce false intrusion alarms and this is essential to its effective implementation.

Information Derived

This methodology allowed a comprehensive investigation on the detection effectiveness because the effectiveness of each model to detect malicious traffic was quantified. The performance aspects were obtained in real-time by the analysis of the latency and throughput. The comparative analysis of the Random Forest and LSTM models gave information on the trade-offs between accuracy and inference speed, allowing to design the most appropriate model to use to detect network intrusion in real time.

IV. Results and Analysis

Comparison of Detection Accuracy.

The accuracy of the used models in detecting malicious network traffic was checked to test their efficiency. LSTM model was able to achieve higher classification power than the Random Forest model. The reason behind this improvement is that LSTM has the capability to acquire temporal dependencies and sequential patterns of network traffic data which is likely to showcase advanced intrusion behaviors.

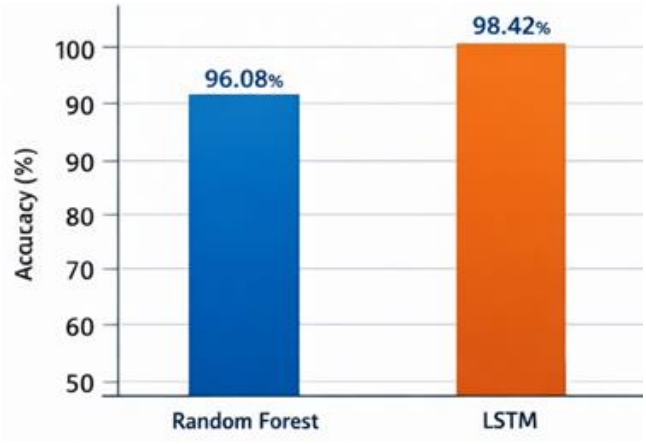

Figure 3: Comparison of Detection Accuracy of Implemented models.

The figure shows the relative accuracy of the random forest and LSTM models in detection. Random Forest model gave an accuracy of 96.08% and the LSTM model gave 98.42% which means that there was a great increase in the performance. Results verify that deep learning models deliver increased detection incidents of sophisticated and dynamic network attacks.

Latency Analysis of Detection in real-time.

The latency in the detection is a crucial element of real-time intrusion detection system because any delay in detection may lead to an attack. The two implemented models were tested on continuous traffic conditions to test the average inference time per record.

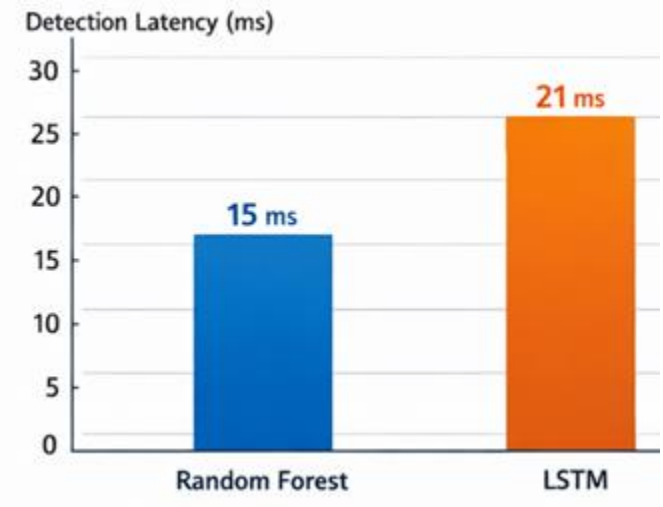

Figure 4: Detection Latency Analysis in real time.

The figure shows how much time is on average spent on the detection latency in the process of inference in real-time. Random Forest model had a lower latency of 15 ms as compared to the LSTM model of 21 ms. Even though the LSTM model is characterized by the high latency of the sequential nature of its processing, the latency is within the acceptable factors of real-time implementation.

Datasets Characteristics of features.

Model performance interpretation requires the understanding of the nature of the data set. The feature set chosen is flow-level statistics, protocol attributes, and time behavior indicators, which are applicable in intrusion detection.

Feature Name	Feature Type	Description
Flow Duration	Numerical	Total duration of the network flow in milliseconds
Protocol Type	Categorical	Network protocol used (e.g., TCP, UDP, ICMP)
Source Bytes	Numerical	Number of bytes sent from source to destination
Destination Bytes	Numerical	Number of bytes sent from destination to source
Packet Count	Numerical	Total number of packets in the flow
Average Packet Size	Numerical	Mean size of packets within the flow
Flow Rate	Numerical	Number of packets transmitted per second
Flag Count	Numerical	Number of TCP control flags set during the flow
Header Length	Numerical	Total header length of packets in the flow
Traffic Label	Categorical	Class label indicating normal or malicious traffic

Table 1: Data Characteristics Information in the Present Study.

The table gives the list of the features of network traffic applied in this study and their description. These characteristics have been chosen in terms of relevance and contribution to detection accuracy to provide the ability to learn effectively whilst reducing computational overhead.

Indeed, this is a model hyperparameter configuration.

The hyperparameters are important in determining model performance. Both models were streamlined in order to balance detectives with computing speed.

Model	Hyperparameter	Value
Random Forest	Number of Trees	200
Random Forest	Maximum Tree Depth	20
Random Forest	Split Criterion	Gini Impurity
Random Forest	Minimum Samples per Leaf	2
LSTM	Number of LSTM Layers	2
LSTM	LSTM Units	64, 32
LSTM	Optimizer	Adam
LSTM	Learning Rate	0.001
LSTM	Batch Size	64
LSTM	Epochs	50

Table 2: Hyperparameter Settings of Implemented Models

The table provides an overview of the optimal values of the hyperparameters of the randomly forest and LSTM models, i.e. the parameters that should be applied to the training and real-time inference processes.

Comparison of Performance Metrics.

To compare the implemented models, a full assessment was done based on standard classification and real-time performance measures.

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	Detection Latency (ms)	False Alarm Rate (%)
Random Forest	96.08	95.74	96.21	95.97	15	3.92
LSTM	98.42	98.11	98.67	98.39	21	1.58

Table 3: Performance Metrics Comparison Across Models

The table is a comparison of the accuracy, precision, recall, F1-score, detection latency and false alarm rate of the two models. The LSTM model performed better than the Random Forest model in all the classification measures whereas the inference latency of the Random Forest model was also lower.

False Negative and False Positive Analysis.

False positives and negative have a direct impact on reliability and usability of intrusion detection systems. It is crucial that the operational environments are designed in such a way that they reduce false alarms but not the detection rates.

Actual \ Predicted	Normal	Malicious
Normal	73,214	1,176
Malicious	1,168	72,959

Table 4: Confusion Matrix Results for the Best-Performing Model

The confusion table is provided in the table with the low false-positive and false-negative error rates. The findings suggest that the LSTM model can provide reliable intrusion detection and low misclassification, which explains why it is applicable to real-time implementation.

General Interpretation and Modelling Choice.

The comparative analysis shows that the LSTM model offers the best combination of detection using and real-time performance. Although the Random Forest model is faster to infer with, the high accuracy of the model plus the lower rates of false alarms indicated by the LSTM model are more consistent with the implementation goals of the proposed study. The outcomes confirm the efficiency of the AI-based intrusion detection systems with real-time threat detection in the current network.

V. CONCLUSION

The findings of the experiments prove that AI-aided intrusion detection has a much higher accuracy and robustness in detection than conventional rule-based methods. It has been found that both of the implemented models can be used to detect malicious network activities successfully, with specific performance features. Random Forest showed high detection and comparably low amounts of computation and inference latency, which rendered the algorithm to be easily applicable to scenarios where fast response and resource utilization are vital. Conversely, LSTM model performed better than the Random Forest model in all the classification metrics such as accuracy, precision, recall and F1-score. The high performance of the LSTM model can be explained by the fact that this model is able to learn temporal and sequential patterns of network traffic that is mostly related to the complex and stealthy attack patterns.

In the real-time performance perspective, detecting latency was a major assessment factor. Even though the LSTM model showed higher inference latency as compared to the Random Forest model, the measured latency was still within reasonable bounds to be widely applicable in intrusion detection in real time. This shows that deep learning models can efficiently be used in real-time security systems under the right optimization. Trade-off between detection accuracy and inference speed underscores the significance of the choice of the models given a certain deployment need.

This work can be relevant to the area of network security as it offers an implementation-oriented assessment of AI-based intrusion detection with real data and standard evaluation metrics. The

comparative study of machine learning and deep learning models under the same experimental conditions can provide an operationally useful experience of their strong and weak aspects. Additionally, the fact that many common tools and models were used indicates that it is possible to implement AI-supported IDS at the expense of technologies that are readily available.

REFERENCES

1. Milyakina, A. (2018). Rethinking literary education in the digital age. *Σημειωτκή-Sign Systems Studies*, *46*(4), 569-589.
2. Nwankwegu, J. A. (2021). DIGITAL TRANSFORMATION AND DIGITAL HUMANITIES: FOCUS ON LANGUAGE AND LITERARY STUDIES. *Nigerian Journal of Arts and Humanities (NJAH)*, *1*(1).
3. Agrawal, S. (2024). Teaching Literature in the Age of Digital Humanities. *EMERGING PARADIGM: INNOVATIONS AND INSIGHT IN ENGLISH LITERATURE AND LANGUAGE RESEARCH IN THE DIGITAL AGE*, 39.
4. London, J. A. (2016). Re-imagining the Cambridge School in the Age of Digital Humanities. *Annual Review of Political Science*, *19*(1), 351-373.
5. Wang, N. (2022). The rise of a new paradigm of literary studies: The challenge of digital humanities. *New Techno Humanities*, *2*(1), 28-33.
6. Chow, R. (2000). *Modern Chinese literary and cultural studies in the age of theory: Reimagining a field*. Duke University Press.
7. Papa, E. (2025). (Re) Thinking Literary Interpretation in the Digital Age: AI, Virtual Reality, and Immersive Reading. *Open Journal of Social Sciences*, *13*(4), 46-67.
8. Kumari, B. (2025). REIMAGINING CREATIVITY: LITERATURE IN THE DIGITAL AGE. *Siddhanta's International Journal of Advanced Research in Arts & Humanities*, 25-32.
9. Berry, D. M., & Fagerjord, A. (2017). *Digital humanities: Knowledge and critique in a digital age*. John Wiley & Sons.
10. Kwong, S. Y. (2025). Reimagining Modernity: Transcultural Narratives and the Reconstruction of Identity in Contemporary Global Literature. *Literary Horizons Review*, *1*(2), 1-8.
11. Melnyk, O., Kyselova, I., & Tereshchuk, M. (2025). The Transformation of Literary Genres in the Digital Age. *Studia Philologica*, (2 (25)), 220-233.

12. Alhinai, M., & Al-Belushi, M. A. K. (2026). Reimagining the Knowledge Economy: A Critical Call to Re-Center the Humanities in Omani Higher Education. *Journal of Arabian Studies*, 1-18.

13. Zafar, Q., Gohar, S. G. A., Jawad, M., & Soomro, M. (2024). The Implications of Digital Culture for Contemporary Literature: Navigating new Narratives, Forms, and Reader Engagement in the Digital Age. *Review of Education, Administration & Law*, *7*(4), 267-286.

14. Hammond, A. (2016). *Literature in the digital age: An introduction*. Cambridge University Press.

15. Risam, R. (2018). *New digital worlds: Postcolonial digital humanities in theory, praxis, and pedagogy*. Northwestern University Press.

16. Batista, E. R. (2026). Reimagining alternative socioecological futures: Transformative narratives for a Second Scientific Revolution. *Environmental Science & Policy*, *176*, 104319.

17. Raab, N. A. (2020). *The humanities in transition from postmodernism into the digital age*. Routledge.

18. Loizeaux, E. B., & Fraistat, N. (Eds.). (2002). *Reimagining textuality: Textual studies in the late age of print*. Univ of Wisconsin Press.

19. Cannon, M., & Potter, J. (2019). Pedagogies of production: Reimagining literacies for the digital age. In *The Palgrave handbook of children's film and television* (pp. 435-450). Cham: Springer International Publishing.

20. Atikurrahman, M. (2025). Reimagining textuality: Digital convergence and literary adaptation in Indonesia. *Lingua Technica: Journal of Digital Literary Studies*, *1*(1), 63-71.

21. Cuff, D., Loukaitou-Sideris, A., Presner, T., Zubiaurre, M., & Crisman, J. J. A. (2020). *Urban humanities: New practices for reimagining the city*. MIT Press.

22. Mentz, S. (2009). Toward a blue cultural studies: The sea, maritime culture, and early modern English literature. *Literature Compass*, *6*(5), 997-1013.