Hybrid Deep Learning Neural Network for Optimizing Video Streaming Traffic Prediction over a Content Delivery Network

Hamza Majid Hamid
Department of Computer Science, Informatics Institute for Postgraduate Studies, Information Technology & Communication University
Baghdad, Iraq
ms202320747@iips.edu.iq

Rana Fareed Ghani
Department of Computer Science
University of Technology
Baghdad, Iraq
Rana.f.ghani@uotechnology.edu.iq

***Abstract*—The Internet has transformed significantly in recent years, giving rise to innovative applications focused on digital content consumption. Advanced frameworks that integrate Content Delivery Networks (CDNs) and Artificial Intelligence (AI) have emerged to optimize content delivery and enhance user experiences. These frameworks use deep learning models to improve traffic prediction, bandwidth management, and network optimization for video streaming. The proposed system preprocesses video data and analyzes traffic patterns during peak hours, identifying underutilized channels and redirecting traffic to less congested routes. This ensures faster content delivery, reduces latency, and prevents bottlenecks during high-demand periods. A Hybrid CNN–LSTM deep learning model is designed to enhance traffic prediction for video streaming over CDNs. By combining the spatial learning capabilities of Convolutional Neural Networks (CNNs) with the temporal modeling strength of Long Short-Term Memory (LSTM) networks, the architecture captures complex spatiotemporal traffic patterns. The model accurately classifies network states, such as depletion, filling, and stalling, achieving a classification accuracy of 99.99%. Its strong generalization ability and low latency make it a scalable and intelligent solution for optimizing video content delivery and improving user experience in dynamic network conditions.**

Keywords*—*Content Delivery Networks (CDNs), Deep Learning (DL), Video Streaming Traffic Prediction, Quality of Experience (QoE)

I. Introduction

Nowadays, Content Delivery Networks (CDNs) and Artificial Intelligence (AI) are essential for managing growing data traffic demands. The integration of CDNs with deep learning significantly enhances user experience, especially for latency-sensitive applications like video streaming and gaming, by optimizing data routing and reducing latency. Deep learning improves content caching, load balancing, and traffic distribution, preventing bottlenecks by directing requests to the nearest available resources.

The Video-on-Demand market is projected to reach USD 212.18 billion by 2030, with digital video accounting for approximately 82% of internet traffic due to the shift towards on-demand viewing [1]. Platforms are adopting hybrid revenue models while investing in localized content to boost engagement.

CDNs rely on caching and intelligent data distribution, making them scalable to meet evolving performance demands. However, video delivery networks face challenges such as latency, bandwidth consumption, and high resource demands. Deep learning enhances CDN efficiency by predicting traffic patterns, optimizing resource allocation, and improving Quality of Experience (QoE). With the rapid expansion of internet usage and streaming services, deep learning-driven CDNs will play a crucial role in ensuring seamless, high-quality content delivery. Dynamic Adaptive Streaming over HTTP (DASH) has emerged as the premier method for its flexibility and efficiency, representing a cornerstone for improved user experiences [2] Incremental statistical analyses of CDN logs, combined with deep learning techniques, provide lightweight evaluations of user experiences, enabling more effective content delivery strategies [3] Moreover, network traffic classification plays a critical role in enhancing QoE by accurately identifying traffic patterns, allowing for better optimization.

Deep learning models are increasingly addressing the complexities of encrypted video streaming and adaptive protocols like MPEG-DASH, offering precise QoE estimations, including video resolution and playback interruptions [4]. Advanced architectures such as CNNs, LSTMs, and Transformers improve traffic prediction, session management, and protocol adaptation. Additionally, GRU-based bandwidth prediction systems dynamically adjust bitrates, optimizing the balance between video quality and stability to minimize buffering and enhance user satisfaction [5].

Mobile live streaming services leverage deep learning-driven predictive analytics and adaptive bitrate (ABR) techniques to reduce latency and enhance video stability [6]. The BANQUET algorithm, enhanced with deep learning, intelligently selects bitrates based on real-time network conditions [7]. Meanwhile, NetScrapper, an AI-powered classifier, surpasses traditional traffic classification methods by integrating with deep learning models for more accurate and real-time assessments [8]. The DASH framework, combined with deep learning-based transport layer analysis, further aids in detecting and mitigating packet interruptions [9], [10], [11].

Deep learning continues to transform adaptive video broadcasting by analyzing user behavior, predicting interposition times, and enhancing overall streaming performance [12], [13]. Fog computing, integrated with deep learning, acts as a crucial intermediary between cloud services and end-users, reducing latency and optimizing bandwidth utilization for superior live streaming experiences [14]. A newly proposed deep learning-based video QoE estimation metric, leveraging pixel-based and network variables, addresses packet loss and delay issues, delivering precise evaluations without requiring original video data [15].

Systems like SENSEI set new standards by integrating deep learning-based adaptive bitrate algorithms with crowdsourcing, leading to substantial QoE improvements [16], [17]. Additionally, a novel database cataloging various stalling patterns and user evaluations plays a key role in refining deep learning-driven QoE prediction models and improving network management strategies [18]. AI-driven predictive analytics, particularly deep learning models, are revolutionizing network performance and content delivery, shaping the future of adaptive video streaming [19].

Hybrid CDN-P2P frameworks leverage deep learning for optimized peer selection, reducing reliance on ISPs and traditional geographical methods [20]. Predictive models, particularly those trained on datasets like the LIVE-Netflix QoE Database, demonstrate significant advantages over conventional metrics, enabling perceptually driven network strategies for superior video quality [21], [22], [23]. Machine learning-based predictive prefetching, enhanced with deep learning techniques in MEC-enabled networks, improves cache efficiency and minimizes access delays by anticipating segment requests. Analysis of segment fetch times and throughput prediction techniques, incorporating player-specific features, is paving the way for robust strategies that promise a seamless video streaming experience.

Deep learning (DL) in traffic management effectively predicts and manages increases in traffic from platforms like (YouTube, Netflix, and gaming). A new CDN model addresses sudden demand surges while maintaining broadcast quality and reducing latency. The Optimizing Content Delivery AI identifies efficient routes for content delivery and balances demand across CDN servers to avoid bottlenecks. The results have shown significant improvements in streaming quality. The research is structured as follows: Section II includes Related Work, Section III presents the work proposal, Section IV presents the results, Section and Section V concludes the study.

II. Related Work

A. Evolution of Content Delivery Networks (CDN) Using Deep Learning

Several researchers examined the evolution of CDN using artificial intelligence, and below, we listed a number of previous works. Live streaming services leveraged AI and CDN technologies to analyze and optimize traffic behavior, ensuring seamless delivery and minimal latency. Machine learning models predicted traffic patterns, enabling efficient resource allocation and enhancing the user experience for live video streaming. In [15], This study proposed a Priority Weighted Round Robin (PWRR) scheduling algorithm within a fog computing architecture to enhance video streaming performance, especially for live video. The architecture simulates client-server interactions using iFogSim, employing DASH and JPEG compression to adapt to bandwidth fluctuations. The dataset included four types of video streams (Lecture, Movie, Live, Conference), each tested across multiple bandwidth conditions. Results showed that PWRR significantly reduced latency and improved video quality compared to traditional WRR, especially under real-time constraints. RMSE analysis confirmed better video fidelity for real-time content with PWRR, validating its suitability for delay-sensitive applications. In [5], To network traffic classification techniques including port-based, Deep Packet Inspection (DPI), statistical machine learning, and deep learning methods. The study compares workflows and highlights that supervised ML models like C4.5 and Random Forest achieved up to 99% accuracy, while CNN-based deep learning models reached over 98% accuracy without manual feature engineering. Public datasets such as Moore, MAWI, UNSW, and ISCX were analyzed, covering both encrypted and non-encrypted traffic from real applications. Semi-supervised models improved labeling efficiency and classification performance on unknown or zero-day traffic. The paper also discusses challenges like encrypted traffic, high computational cost, and privacy concerns, and proposes future directions for scalable and real-time classification solutions. In [24], hybrid early traffic classifier combining unencrypted TLS handshake features with statistical flow-based attributes (e.g., packet sizes and inter-arrival times) to address classification challenges under Encrypted ClientHello (ECH). They constructed a diverse multi-country TLS dataset of over 600,000 labeled flows across 19 traffic classes including VOD and live streaming services. The dataset simulates realistic ECH encryption scenarios by masking TLS metadata such as SNI. hRFTC achieved up to 94.6% macro F-score, significantly outperforming state-of-the-art packet-only and hybrid classifiers. Their findings highlight the importance of regional retraining due to geo-dependent traffic patterns and support the value of hybrid modeling for encrypted traffic analysis. in [25], context-aware adaptive prefetching system for DASH video streaming over 5G networks using Multi-access Edge Computing (MEC). They trained four ML classification models (RF, KN, SVM, LDA) on a custom DASH session dataset collected via 20 GStreamer-based clients, selecting the Random Forest model for its highest accuracy of 78.1%. The model predicted the next video segment bitrate, enabling proactive caching at the MEC. Their "Predictive Cache" strategy achieved a 37.25% data traffic reduction and QoE improvement (avg. 4.31), nearly matching the best-case "Preemptive Cache" (QoE 4.38) with lower resource use. The method outperformed legacy caching in terms of both network efficiency and end-user experience. in [26], a novel system to identify encrypted live streaming channels using traffic patterns of time-synchronized (time-sync) comments rather than traditional bitrate analysis. They developed inter- and intra-application traffic filters using SNI, IP headers, and a CNN model, followed by comment rate estimation via least squares, and delay-tolerant similarity matching using improved DTW and SVM classification. The dataset was captured from YouTube and BiliBili live streams over six hours using AWS EC2 instances simulating MITM attacks. Their method achieved 93.2% accuracy in traffic filtering, 91% in comment rate estimation, and up to 98.2% overall matching accuracy after 500 seconds of eavesdropping, outperforming prior bitrate-based methods. The system demonstrated resilience to bandwidth fluctuations and mixed traffic environments, proving robust and scalable for real-world deployment.

B. Content Delivery Networks

CDNs use distributed servers to deliver content quickly through the nearest node to the user, reducing the load on the origin server and improving performance. Users are directed to the closest server based on their location and server load to ensure speed and efficiency. CDNs provide fast response times and handle sudden traffic surges by dynamically distributing the load [27]. CDNs are essential in modern web architectures, improving performance by caching frequently accessed content across globally distributed edge nodes (PoPs). Key concepts include content caching, regulated by TTL to determine how long resources stay cached; purging, which updates cached content from origin servers; multiple origins, such as cloud storage or dedicated servers; and access restrictions, controlling who can access cached content based on domains, regions, or IP groups. By integrating AI, CDNs can further optimize caching, reduce latency, enhance availability, personalize content delivery, and improve user experience, making content delivery faster, more reliable, and efficient [28].

III. METHODOLOGY

To enhance CDN performance and ensure high-quality video streaming, this research employs a hybrid deep learning neural network capable of capturing complex spatiotemporal traffic patterns [29]. By leveraging historical traffic data and user interaction behavior, the proposed deep learning model accurately predicts network conditions and anticipates fluctuations in demand. This enables proactive resource allocation and dynamic adjustment of streaming parameters such as bitrate, resolution, and buffer size[30]. In addition, real-time monitoring mechanisms feed into the model to support adaptive decision-making for optimized content placement and reduced latency, particularly during peak traffic periods. The deep learning-driven methodology ensures efficient bandwidth utilization, uninterrupted playback, and consistent quality of experience (QoE). Ultimately, this approach maximizes the operational efficiency of the CDN infrastructure while significantly improving end-user satisfaction.

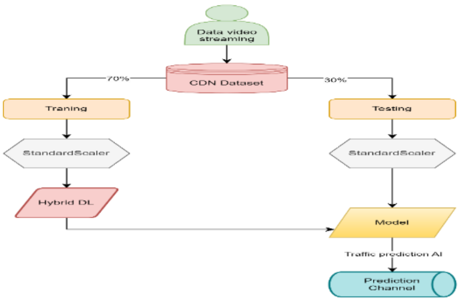

Fig. 1. Traffic Prediction System for Video Streaming Using AI and Deep Learning.xample of a figure caption

A. Dataset

In [31], a comprehensive dataset comprising 4.5 million entries was gathered from 45 days of continuous video streaming via the original YouTube mobile app. It included 11,142 measurements across 171 bandwidth entries and 80 diverse network conditions, totaling 332GB of video traffic over TCP and UDP/QUIC protocols. Covering extensive real-world mobile streaming scenarios, this dataset enables accurate modeling and prediction of modern network behaviors. Data was systematically divided into 70% for training and 30% for testing, facilitating robust machine learning model development and evaluation. The dataset classifies network behavior as "filling", "depletion," and "stalling".

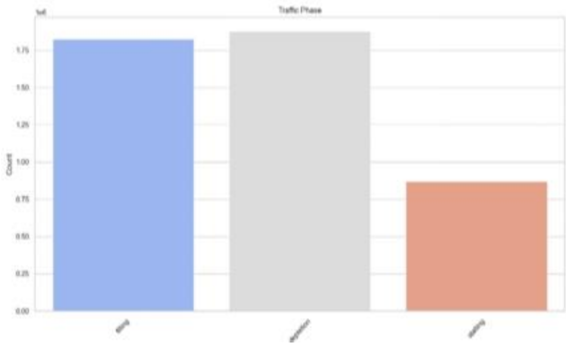

Fig. 2. Distribution of Network Traffic States

The figure shows the distribution of network traffic states, with "depletion" being the most frequent, followed by "filling," and "stalling" being the least common.

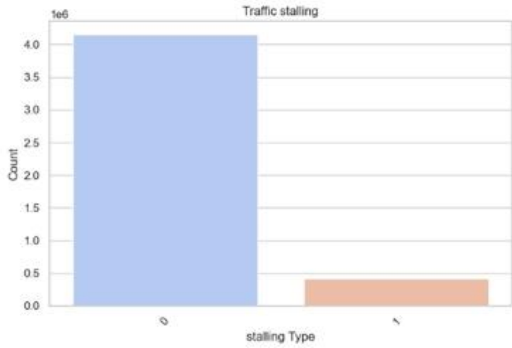

Fig. 3. Distribution of Traffic Stalling Types

The figure illustrates the distribution of stalling in network traffic, highlighting that type "0" significantly dominates over type "1."

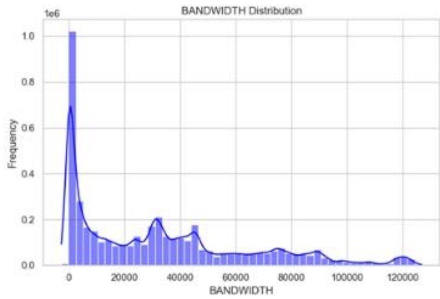

Fig. 4. Bandwidth Distribution

The figure shows the distribution of bandwidth values, where low bandwidth ranges (below 10,000) dominate the dataset, with frequency decreasing as bandwidth increases.

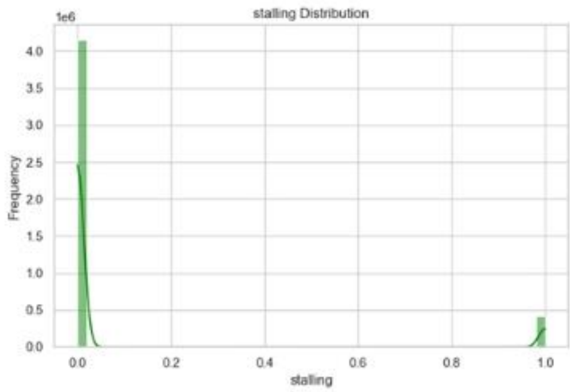

Fig. 5. Stalling Value Distribution

The figure reveals that most stalling values are concentrated near 0, indicating minimal interruptions, while a smaller peak near 1 suggests fewer instances of complete stalling.

B. Data Normalization Using Standard Scaler

In [32], prior to training the deep learning model, all input features were normalized using the Standard Scaler technique to enhance training efficiency and stability. This transformation standardizes each feature by removing the mean and scaling to unit variance, ensuring that all input variables contribute proportionally during the learning process. Mathematically, for each feature xi, the standardized value xi is computed as:

$$x^{-}_{i} = \frac{x_i - \mu}{\sigma}$$

where: x^{-}_{i} is the normalized value; μ represents the mean of the feature; σ denotes the standard deviation.

This normalization is particularly important in deep learning architectures, as it accelerates the convergence of gradient-based optimizers and prevents features with larger magnitudes from dominating the learning process. By transforming the data to follow a standard normal distribution, the Standard Scaler helps maintain numerical stability across layers and improves generalization in models such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs), including LSTM layers.

C. Deep learning

To develop an effective deep learning-based model for intelligent traffic prediction, we designed a hybrid architecture that combines one-dimensional convolutional neural networks (1D-CNNs) with Long Short-Term Memory (LSTM) layers[33]. This approach leverages the feature extraction capabilities of CNNs and the temporal learning ability of LSTMs to model complex spatiotemporal traffic patterns[34]. The network was implemented using the Keras framework with a TensorFlow backend.

The architecture begins with a Conv1D layer consisting of 16 filters, a kernel size of 3, and "same" padding to preserve input dimensions. This layer is followed by a LeakyReLU activation function with an alpha coefficient of 0.3 to introduce non-linearity while mitigating the vanishing gradient problem. A MaxPooling1D layer with a pool size and stride of 1, and "same" padding, is applied to downsample the feature maps while retaining spatial resolution.

The convolutional block is repeated with 32 and then 64 filters, each time using a kernel size of 3, maintaining consistent padding and stride. After each convolutional layer, the non-linearity is maintained using LeakyReLU activations. Selective pooling is applied to capture salient features while reducing computational complexity. The network depth is gradually increased and then symmetrically decreased (i.e., 64 → 32 → 16 filters) to capture both high- and low-level features effectively.

To incorporate temporal dependencies in the sequence data, two LSTM layers are integrated at key points in the architecture. The first LSTM layer includes 16 units and is configured to return sequences, which allows the retention of temporal states for downstream layers. A second LSTM layer with 32 units is employed later in the architecture to enhance the model's memory of time-dependent patterns.

The later layers include additional Conv1D and MaxPooling1D operations to refine the learned features. The final Conv1D layer applies 35 filters with a linear activation function to prepare the sequence output for flattening. The Flatten layer transforms the multi-dimensional output into a one-dimensional vector, which is passed to a Dense layer with three output units and a softmax activation function for multi-class classification.

The complete model comprises 31,747 parameters, all of which are trainable. No non-trainable parameters are present. This structure ensures a balanced trade-off between model complexity and generalization ability.

- Mathematical Formulation of the Proposed Deep Learning Model

Equation (1) represents the overall forward pass of The hybrid CNN–LSTM model can be described as a hierarchical mapping function that transforms the input multivariate time series , where T s the sequence length and F the number of features (in this case, F=1), into a probability distribution over target classes for multi-class classification.

The forward pass of the model is formally represented as:

$$\hat{y} = \text{Softmax}\left(W_d \cdot \text{Flatten}\left(\Phi_{\text{LSTM}_2}\left(\Phi_{\text{CNN+LSTM}_1}(X)\right)\right) + b_d\right) —— (1)$$

where: $\Phi_{\text{CNN+LSTM}_1}(\cdot)$: Composite function representing the first series of convolutional, LeakyReLU, and pooling layers, followed by the first LSTM layer with return sequences; $\Phi_{\text{LSTM}_2}(\cdot)$: Second LSTM layer processing the spatiotemporal features from earlier layers; $\text{Flatten}(\cdot)$: Reshaping operation converting tensor output into a vector; $W_d,\ b_d$: Weights and biases of the final dense (fully connected) classification layer; $\text{Softmax}(\cdot)$: Activation function to produce class probabilities.

Equation (2) defines the output of each Conv1D layer, where a one-dimensional convolution is applied followed by a non-linear activation function

$$\mathrm{X}_{\text{out}}^{(l)} = \mathrm{f}\left(\mathrm{W}^{(l)} * \mathrm{X}_{\text{in}}^{(l)} + \mathrm{b}^{(l)}\right) —— (2)$$

Equation (3) shows the LeakyReLU activation function used to introduce non-linearity while avoiding the dying ReLU problem:

$$f(x) = \begin{cases} x, if\ x > 0 \\ \alpha x, otherwise \end{cases} \quad with\ \alpha = 0.3 —— (3)$$

Equation (4) expresses the transformation within each LSTM cell, capturing long-term dependencies through its memory mechanism:

$$h_t, c_t = \text{LSTM}(x_t, h_{t-1}, c_{t-1}) —— (4)$$

where: h_t: Hidden state; c_t: Cell state at time step t; x_t: Input at time step t.

A. *Advantages of the Proposed Model*

- Efficient Spatiotemporal Feature Extraction

The integration of 1D-CNN layers enables automatic extraction of local temporal patterns in the input sequence, capturing both short-term fluctuations and trend features effectively.

- Temporal Dependency Modeling

The use of LSTM layers allows the model to retain long-term dependencies in sequential data, which is crucial for accurately forecasting time-series patterns such as traffic behavior.

- Balanced Pooling Strategy

Carefully designed MaxPooling layers reduce computational overhead while preserving important features, maintaining a balance between model performance and training efficiency.

- Multi-Class Prediction Capability

The final Dense layer with softmax activation supports multi-class classification tasks, enabling the model to distinguish between multiple traffic states or usage conditions.

- Scalability and Modularity

The hybrid structure can be easily extended or modified to adapt to other time-series prediction tasks, making it a flexible framework for a wide range of applications.

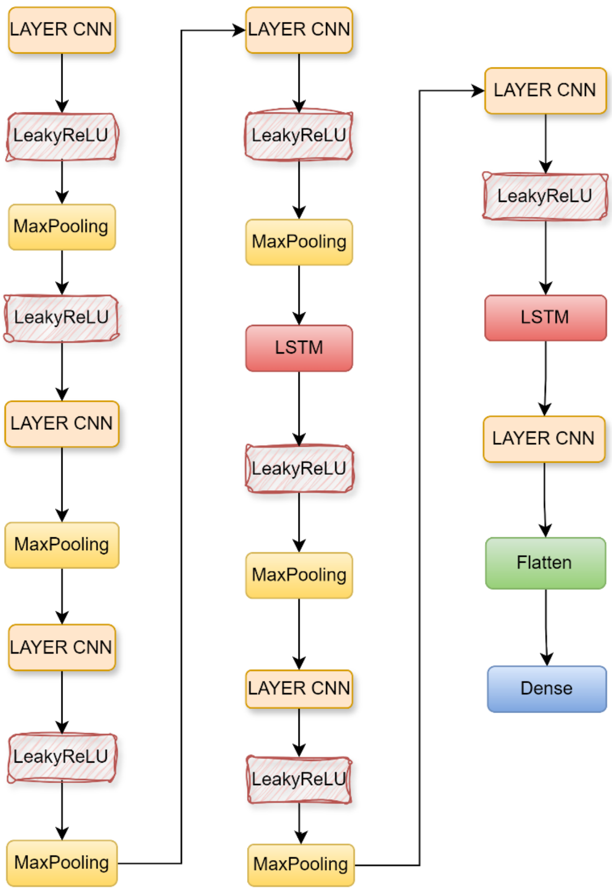

Fig. 6. Architecture of the Hybrid CNN–LSTM Model of Layer

Fig. 6 illustrates the detailed architecture of the proposed hybrid CNN–LSTM model. The network begins with several one-dimensional convolutional (Conv1D) layers interleaved with LeakyReLU activation functions and MaxPooling1D layers to extract robust multi-scale spatial features. Notably, the early convolutional layers progressively increase the number of filters (16 → 32 → 64), allowing hierarchical feature representation.

Following the convolutional blocks, LSTM units are introduced to capture the temporal dependencies inherent in sequential traffic data. Two LSTM layers are strategically placed after feature extraction stages to retain long-term contextual patterns across time steps. The architecture also utilizes additional convolutional layers post-LSTM to further refine feature representations. A Flatten layer transitions the three-dimensional output to a two-dimensional format suitable for a Dense output layer that performs the final classification.

This layered configuration results in a total of approximately 28,747 trainable parameters, optimizing both spatial and temporal learning in a compact and computationally efficient manner. As shown in Table 1, the hybrid model significantly outperforms baseline models across all evaluation metrics — including precision, recall, F1-score, and accuracy — affirming its suitability for complex sequence classification tasks such as network traffic state prediction.

TABLE I. STRUCTURAL DESIGN OF THE DISTRIBUTED HYBRID CNN–LSTM NETWORK

Layer (type)	Output Shape	Param
conv1d_1 (Conv1D)	(None, 9, 16)	64
leaky_re_lu_1 (LeakyReLU)	(None, 9, 16)	0
max_pooling1d_1 (MaxPooling1)	(None, 9, 16)	0
leaky_re_lu_2 (LeakyReLU)	(None, 9, 16)	0
conv1d_2 (Conv1D)	(None, 9, 32)	1568
max_pooling1d_2 (MaxPooling1)	(None, 9, 32)	0
conv1d_3 (Conv1D)	(None, 9, 64)	6208
leaky_re_lu_3 (LeakyReLU)	(None, 9, 64)	0
max_pooling1d_3 (MaxPooling1)	(None, 9, 64)	0
conv1d_4 (Conv1D)	(None, 9, 32)	6176
leaky_re_lu_4 (LeakyReLU)	(None, 9, 32)	0
max_pooling1d_4 (MaxPooling1)	(None, 9, 32)	0
lstm_1 (LSTM)	(None, 9, 16)	3136
leaky_re_lu_5 (LeakyReLU)	(None, 9, 16)	0
max_pooling1d_5 (MaxPooling1)	(None, 9, 16)	0
conv1d_5 (Conv1D)	(None, 9, 16)	784
leaky_re_lu_6 (LeakyReLU)	(None, 9, 16)	0
max_pooling1d_6 (MaxPooling1)	(None, 5, 16)	0
conv1d_6 (Conv1D)	(None, 5, 32)	1568
leaky_re_lu_7 (LeakyReLU)	(None, 5, 32)	0
lstm_2 (LSTM)	(None, 5, 32)	8320
conv1d_7 (Conv1D)	(None, 5, 35)	3395
flatten_1 (Flatten)	(None, 175)	0
dense_1 (Dense)	(None, 3)	528

IV. EXPERIMENTAL RESULTS

In this study, we evaluate the performance of deep learning models for intelligent traffic prediction using a real-world CDN YouTube dataset related to mobile streaming behavior. The experimental framework implements and compares multiple deep learning architectures, including the proposed hybrid CNN–LSTM model. Model performance is assessed using standard classification metrics—Precision [35], Recall [36], F1-score [37], Accuracy [38], and Execution Time—which provide comprehensive insights into both predictive quality and computational efficiency.

The evaluation metrics are formally defined as follows:

(5) represents the precision metric, which quantifies the proportion of correctly predicted positive cases among all instances classified as positive:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{5}$$

(6) defines recall, which measures the model's ability to correctly identify all actual positive cases in the dataset:

$$\text{Recall} = \frac{TP}{TP+FN} \tag{6}$$

(7) expresses the F1-score, calculated as the harmonic mean of precision and recall. It provides a balanced assessment of the model's performance, especially when class distributions are imbalanced:

$$F_1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \text{———} (7)$$

(8) denotes the accuracy metric, which reflects the overall correctness of the classification model by evaluating the ratio of all correctly predicted observations to the total number of instances:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \text{———} (8)$$

where TP (True Positives), FP (False Positives), FN (False Negatives), and TN (True Negatives) represent the standard components of the confusion matrix.

The models were trained and evaluated using a train-test split approach to ensure robustness and generalization. Execution time was measured to assess the computational load associated with each architecture. The experimental findings indicate that the proposed CNN–LSTM model outperforms alternative deep learning baselines in terms of accuracy and F1-score, effectively capturing both spatial and temporal dependencies in the traffic data. In contrast, a lighter deep learning variant (e.g., a CNN-only model) achieves faster execution time due to its reduced sequential complexity.

This observed trade-off between predictive performance and computational efficiency is critically analyzed to determine the optimal model configuration for real-time CDN traffic prediction, where both accuracy and low latency are essential.

TABLE II. DEEP LEARNING

Class	hybrid CNN–LSTM		
	Precision	Recall	F1-Score
Depletion	0.99	0.99	0.99
Filling	0.99	0.99	0.99
Stalling	0.99	0.99	0.99
Accuracy			0.99
Weighted Avg	0.99	0.99	0.99

Fig. 7. Hybrid CNN-LSTM

The chart presents a comprehensive evaluation of the Hybrid CNN–LSTM model's performance in classifying three traffic states: Depletion, Filling, and Stalling. The model demonstrates consistently high scores across key performance metrics—Precision, Recall, and F1-Score—for all three classes, as illustrated in Fig. 7. These results underscore the model’s robustness and effectiveness in handling imbalanced and complex traffic patterns. The average performance values across all metrics approach or reach 100%, highlighting the reliability of the model in predictive accuracy. Furthermore, Table II summarizes these metrics numerically. The model achieves perfect classification for Depletion and Filling, with minimal misclassification in the Stalling category. These findings confirm the model's suitability for intelligent traffic prediction and real-time network optimization.

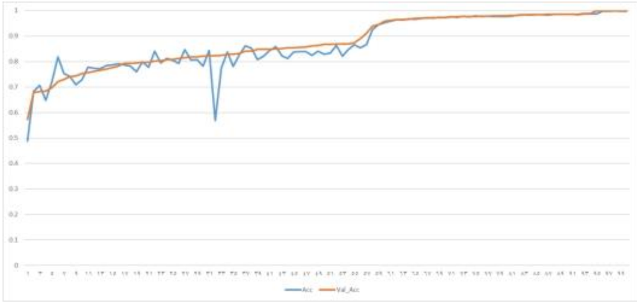

Fig. 8. Performance Convergence of the Hybrid CNN–LSTM Model

Fig. 8 illustrates the progression of accuracy and validation accuracy throughout the model development process. The Hybrid CNN–LSTM model demonstrates a consistent upward trend in performance, with minor fluctuations observed in the initial stages that gradually diminish. As the learning process advances, both training and validation accuracy exhibit strong convergence, ultimately reaching approximately 99.9%. This convergence signifies the model’s robust generalization capability and its effectiveness in capturing complex temporal and spatial dependencies within the dataset.

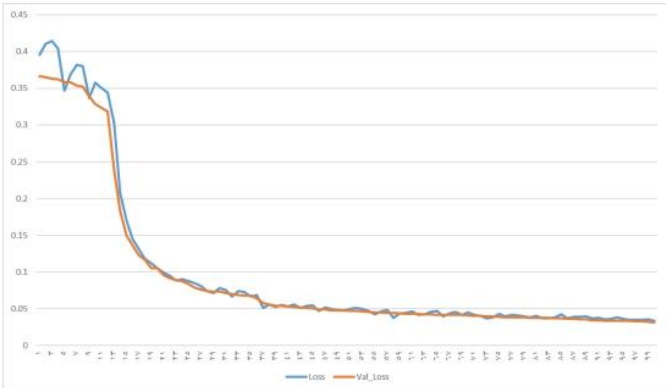

Fig. 9. Convergence Behavior of Loss and Validation Loss for the Hybrid CNN–LSTM Model

Fig. 9 presents the reduction in loss and validation loss during the optimization of the Hybrid CNN–LSTM model. In the initial stages, both curves exhibit higher loss values with minor fluctuations, reflecting the model's adjustment to learning from the data. As training progresses, a sharp and consistent decline is observed, indicating effective minimization of error. Eventually, both training and validation loss stabilize at values close to zero, signifying strong convergence and the model’s ability to capture the underlying data distribution accurately.

V. DISCUSSION

As shown in Table III, a comparative analysis of various deep learning approaches for network performance prediction is presented. The table outlines the evaluation of several models based on essential metrics such as precision, recall, F1-score, and accuracy. Compared to previous works, the proposed Hybrid CNN–LSTM Model in this study demonstrates outstanding performance, reflecting its capability to capture complex temporal and spatial patterns in network data. The model's robustness and consistency across different metrics suggest its effectiveness in real-world scenarios. Notably, the accuracy achieved in this work reaches 99.99%, which highlights the superiority of the proposed approach.

TABLE III. COMPARISON OF DEEP LEARNING METHODS FOR NETWORK PERFORMANCE

Reference	Method	Precision (%)	Recall (%)	F1-score (%)	Accuracy (%)
[43]	ICLSTM - Mixed Encrypted Traffic	98	98.4	98.2	98.2
[44]	CSCNN	95.15	92.01	93.55	97.7
[41]	ReCLive (LSTM + RF)	93	–	90	95
[42]	1D-CNN (spatial only), Bi-LSTM with attention (temporal only)	90.3 91.1	88.4 89.5	89.3 90.3	92
[36]	Batali	95	95	95	95
[39]	ASAPjitter (GRU + CNN), FEATjitter (GRU + MLP)	97.5 91.9	97.5 91.9	97.5 91.9	–
[40]	LSTM	78.42	77.13	–	81.01
[23]	Broadcaster behavior clustering using K-Means	0.85	0.96	0.90	–
This work	**Hybrid CNN–LSTM Model**	**99**	**99**	**99**	**99.99**

VI. CONCLUSIONS

In this study, we proposed a hybrid deep learning architecture that integrates one-dimensional convolutional neural networks (1D-CNNs) with Long Short-Term Memory (LSTM) networks for effective traffic prediction and time-series classification. The model is designed to extract both spatial and temporal features by leveraging the strengths of CNNs in local pattern detection and LSTMs in modeling long-term dependencies.

Through the careful arrangement of convolutional, pooling, and recurrent layers—combined with LeakyReLU activations—the architecture achieves a high level of representational capacity while maintaining computational efficiency with only 31,747 trainable parameters. The hybrid structure enables accurate, robust, and real-time predictions across multi-class scenarios, making it well-suited for intelligent traffic systems and similar time-sensitive applications.

Future work will focus on enhancing the security of AI-powered CDNs using adversarial defenses. These advancements aim to establish a robust, scalable, and privacy-preserving infrastructure for intelligent video delivery.

REFERENCES

[1] "Video-on-Demand Market Analysis | Industry Trends, Size & Growth Outlook." Accessed: Mar. 07, 2025. [Online]. Available: https://www.mordorintelligence.com/industry-reports/video-on-demand-market

[2] A. Hodroj, M. Ibrahim, and Y. Hadjadj-Aoul, "A Survey on Video Streaming in Multipath and Multihomed Overlay Networks," 2021, Institute of Electrical and Electronics Engineers Inc. doi: 10.1109/ACCESS.2021.3076464.

[3] H. Girault, L. Bonniot, and C. Neumann, "Inferring Video Streaming Quality of Experience at Scale using Incremental Statistics from CDN Logs," MHV 2024 - Proceedings of the 3rd Mile-High Video Conference, pp. 34–40, Feb. 2024, doi: 10.1145/3638036.3640803.

[4] I. Orsolic and L. Skorin-Kapov, "A Framework for in-Network QoE Monitoring of Encrypted Video Streaming," IEEE Access, vol. 8, pp. 74691–74706, 2020, doi: 10.1109/ACCESS.2020.2988735.

[5] A. Azab, M. Khasawneh, S. Alrabaee, K.-K. R. Choo, and M. Sarsour, "Network traffic classification: Techniques, datasets, and challenges," 2024, doi: 10.1016/j.dcan.2022.09.009.

[6] T. V. Huu, S. Van Pham, T. N. T. Huong, and H. C. Le, "QoE Aware Video Streaming Scheme Utilizing GRU-Based Bandwidth Prediction and Adaptive Bitrate Selection for Heterogeneous Mobile Networks," IEEE Access, vol. 12, pp. 45785–45795, 2024, doi: 10.1109/ACCESS.2024.3382155.

[7] Z. Li, J. Li, Q. Wu, G. Tyson, and G. Xie, "A Large-Scale Measurement and Optimization of Mobile Live Streaming Services," IEEE Trans Mob Comput, vol. XXX, no. 1, Accessed: Jan. 04, 2025. [Online]. Available: https://www.dropbox.com/s/nmwh75syvi6duxv/MLS-

[8] T. Kimura, T. Kimura, A. Matsumoto, and K. Yamagishi, "Balancing Quality of Experience and Traffic Volume in Adaptive Bitrate Streaming," IEEE Access, vol. 9, pp. 15530–15547, 2021, doi: 10.1109/ACCESS.2021.3052552.

[9] A. A. Ahmed and G. Agunsoye, "A real-time network traffic classifier for online applications using machine learning," Algorithms, vol. 14, no. 8, Aug. 2021, doi: 10.3390/a14080250.

[10] R. X. Zhang et al., "A Practical Learning-based Approach for Viewer Scheduling in the Crowdsourced Live Streaming," ACM Transactions on Multimedia Computing, Communications and Applications, vol. 16, no. 2s, Jul. 2020, doi: 10.1145/3397226.

[11] J. Kim, Y. Jung, H. Yeo, J. Ye, and D. Han, "Neural-Enhanced Live Streaming: Improving Live Video Ingest via Online Learning," in SIGCOMM 2020 - Proceedings of the 2020 Annual Conference of the ACM Special Interest Group on Data Communication on the Applications, Technologies, Architectures, and Protocols for Computer Communication, Association for Computing Machinery, Jul. 2020, pp. 107–125. doi: 10.1145/3387514.3405856.

[12] M. Martinez-Caro and M. D. Cano, "On the identification and prediction of stalling events to improve qoe in video streaming," Electronics (Switzerland), vol. 10, no. 6, pp. 1–17, Mar. 2021, doi: 10.3390/electronics10060753.

[13] F. Loh, F. Poignée, F. Wamser, F. Leidinger, and T. Hoßfeld, "Uplink vs. Downlink: Machine learning-based quality prediction for http adaptive video streaming," Sensors, vol. 21, no. 12, Jun. 2021, doi: 10.3390/s21124172.

[14] R. Farahani, Z. Azimi, C. Timmerer, and R. Prodan, "Towards AI-Assisted Sustainable Adaptive Video Streaming Systems: Tutorial and Survey," Jun. 2024, [Online]. Available: http://arxiv.org/abs/2406.02302

[15] S. F. Hassan and R. FareedGhani, "PWRR algorithm for video streaming process using fog computing," Baghdad Science Journal, vol. 16, no. 3, pp. 667–676, 2019, doi: 10.21123/bsj.2019.16.3.0667.

[16] R. F. Ghani and A. S. Ajrash, "Quality of experience metric of streaming video: A survey," Aug. 30, 2018, University of Baghdad-College of Science. doi: 10.24996/IJS.2018.59.3B.19.

[17] A. S. Ajrash, R. F. Ghani, and L. Al-Jobouri, "Quality of Experience (QoE) Measurement Algorithm for Transmitted Video."

[18] X. Zhang, Y. Ou, S. Sen, and J. Jiang, Open access to the Proceedings of the 18th USENIX Symposium on Networked Systems Design and Implementation is sponsored by SenSei: Aligning Video Streaming Quality with Dynamic User Sensitivity SENSEI: Aligning Video Streaming Quality with Dynamic User Sensitivity. [Online]. Available: https://www.usenix.org/conference/nsdi21/presentation/zhang-xu

[19] D. Ghadiyaram, J. Pan, and A. C. Bovik, "A Subjective and Objective Study of Stalling Events in Mobile Streaming Videos," IEEE Transactions on Circuits and Systems for Video Technology, vol. 29, no. 1, pp. 183–197, Jan. 2019, doi: 10.1109/TCSVT.2017.2768542.

[20] K. Khan, "Enhancing Adaptive Video Streaming Through AI-Driven Predictive Analytics for Network Conditions: A Comprehensive Review," International Transactions on Electrical Engineering and Computer Science, vol. 3, no. 1, pp. 57–68, Mar. 2024, doi: 10.62760/iteecs.3.1.2024.67.

[21] Z. Ma, S. Roubia, F. Giroire, and G. Urvoy-Keller, "When Locality is not enough: Boosting Peer Selection of Hybrid CDN-P2P Live

Streaming Systems using Machine Learning”, Accessed: Jan. 04, 2025. [Online]. Available: https://tools.ietf.org/html/rfc5389

[22] I. Sarkar et al., “Multi-Source HTTP Live Streaming (MSHLS), an ABR Algorithm for Hybrid V2V-CDN Video Streaming.” [Online]. Available: https://hal.science/hal-03452586v1

[23] J. Li, Z. Li, Q. Wu, and G. Tyson, “On Uploading Behavior and Optimizations of a Mobile Live Streaming Service.”

[24] .Shamsimukhametov Danil, Anton Kurapov, Mikhail Liubogoshchev, and Evgeny Khorov. "Early Traffic Classification with Encrypted ClientHello: a Multi-Country Study." IEEE Access (2024).

[25] J. Uriol, I. Yeregui, A. Gabilondo, R. Viola, P. Angueira, and J. Montalban, “Context-Aware Adaptive Prefetching for DASH Streaming over 5G Networks,” in IEEE International Symposium on Broadband Multimedia Systems and Broadcasting, BMSB, IEEE Computer Society, 2023. doi: 10.1109/BMSB58369.2023.10211275.

[26] M. Wang, X. Tang, F. Chen, and Q. Lu, “Encrypted Live Streaming Channel Identification with Time-Sync Comments,” IEEE Access, vol. 10, pp. 27630–27642, 2022, doi: 10.1109/ACCESS.2022.3157716.

[27] “Computer-Networks---A-Tanenbaum---5th-edition”.

[28] X. Nazirjon Nabijanovich, “ICARHSE International Conference on Advance Research in Humanities, Sciences and Education Hosted from Berlin, GERMANY https://confrencea.org September 15 th 2022 THE ROLE OF AI IN CONTENT DELIVERY NETWORKS.” [Online]. Available: https://confrencea.org

[29] Min-max Normalizer -Al-Latief, Shahad Thamear Abd, Salman Yussof, Azhana Ahmad, Saif Mohanad Khadim, and Ahmed Alkhayyat. "WAR Strategy Algorithm-based Hybrid Optimization for Accurate and Rapid Speech Recognition." Iraqi Journal for Computer Science and Mathematics 6, no. 1 (2025): 13.

[30] .Deep-learning-Kadhim, Saif Mohanad, Johnny Ko Siaw Paw, Yaw Chong Tak, Shahad Ameen, and Ahmed Alkhayyat. "Deep Learning for Robust Iris Recognition: Introducing Synchronized Spatiotemporal Linear Discriminant Model-Iris." Advances in Artificial Intelligence and Machine Learning 5, no. 1 (March 25, 2025): 3446–3464.

[31] F. Loh, F. Wamser, F. Poignée, S. Geißler, and T. Hoßfeld, “YouTube Dataset on Mobile Streaming for Internet Traffic Modeling and Streaming Analysis,” Sci Data, vol. 9, no. 1, Dec. 2022, doi: 10.1038/s41597-022-01418-y.

[32] Deep-learning-Abdulhasan, Raed Abdulkareem, Shahad Thamear Abd Al-latief, and Saif Mohanad Kadhim. "Instant learning based on deep neural network with linear discriminant analysis features extraction for accurate iris recognition system." Multimedia Tools and Applications 83, no. 11 (2024): 32099-32122.

[33] Deep-learning-Kadhim, Saif Mohanad, Johnny Koh Siaw Paw, Yaw Chong Tak, and Shahad Ameen. "Deep Learning Models for Biometric Recognition based on Face, Finger vein, Fingerprint, and Iris: A Survey." Journal of Smart Internet of Things (JSIoT) 2024, no. 01 (2024): 117-157.

[34] 34.Deep-learning-Abd Al-Latief, Shahad Thamear, Salman Yussof, Azhana Ahmad, and Saif Khadim. "Deep Learning for Sign Language Recognition: A Comparative Review." Journal of Smart Internet of Things (JSIoT) 2024, no. 01 (2024): 77-116.

[35] .Kadhim, Saif Mohanad, Johnny Koh Siaw Paw, Yaw Chong Tak, and Shahad Thamear Abd Al-Latief. "Instant Fingerprint Recognition Using Optimized Machine Learning Models by Corona Virus Optimization Algorithm." In 2024 International Conference on Intelligent Computing and Next Generation Networks (ICNGN), pp. 01-07. IEEE, 2024.

[36] Kadhim, Saif Mohanad, Johnny Koh Siaw Paw, and Yaw Chong Tak. "An Optimized Machine Learning Models by Metaheuristic Corona Virus Optimization Algorithm for Precise Iris Recognition." Advances in Artificial Intelligence and Machine Learning 5, no. 1 (March 22, 2025): 3389–3408.

[37] .Kadhim, Saif Mohanad, Johnny Koh Siaw Paw, Yaw Chong Tak, Shahad Thamear Abd Al - Latief, Ahmed Alkhayyat, and Deepak Gupta. "Lionfish Search Algorithm: A Novel Nature - Inspired Metaheuristic." Expert Systems 42, no. 4 (2025): e70016.

[38] Abd Al-Latief, Shahad Thamear, Salman Yussof, Azhana Ahmad, Saif Mohanad Khadim, and Raed Abdulkareem Abdulhasan. "Instant Sign Language Recognition by WAR Strategy Algorithm Based Tuned Machine Learning." International Journal of Networked and Distributed Computing 12, no. 2 (2024): 344-361.

[39] Lisas, Tautvydas, and Ruairí de Fréin. "Sequential learning for modeling video quality of delivery metrics." IEEE Access 11 (2023): 107783-107797

[40] Liu, Ke, Kan Wu, Hua Wang, Ke Zhou, Ji Zhang, and Cong Li. "SLAP: An adaptive, learned admission policy for content delivery network caching." In 2023 IEEE International Parallel and Distributed Processing Symposium (IPDPS), pp. 457-467. IEEE, 2023.

[41] S.C. Madanapalli, A. Mathai, H. H. Gharakheili, and V. Sivaraman, “ReCLive: Real-Time Classification and QoE Inference of Live Video Streaming Services,” in IEEE/ACM International Symposium on Quality of Service (IWQOS), Jun. 2021, pp. 1–10, doi: 10.1109/IWQOS52092.2021.9521288.

[42] Seydali, Mehdi, Farshad Khunjush, Behzad Akbari, and Javad Dogani. "CBS: A deep learning approach for encrypted traffic classification with mixed spatio-temporal and statistical features." IEEE Access 11 (2023): 141674-141702.

[43] Lu, Bei, Nurbol Luktarhan, Chao Ding, and Wenhui Zhang. "ICLSTM: encrypted traffic service identification based on inception-LSTM neural network." Symmetry 13, no. 6 (2021): 1080.

[44] Soleymanpour, Shiva, Hossein Sadr, and Mojdeh Nazari Soleimandarabi. "CSCNN: cost-sensitive convolutional neural network for encrypted traffic classification." Neural Processing Letters 53, no. 5 (2021): 3497-3523.