INTERNATIONAL JOURNAL OF INFORMATION AND COMMUNICATION TECHNOLOGY TRENDS

Vol. 6 No. 02 (2026)

e.ISSN: 2409-5222

https://ijictt.ictt.info

Fuzzy-Rule Traffic Predictors for Nonstationary Network Telemetry Streams

Raymond Ka-Man Chau

Department of Electronic and Computer Engineering, School of Engineering, Hong Kong University of Science and Technology, Hong Kong, Hong Kong SAR, China

Abstract

Modern network infrastructures generate continuous, high-velocity telemetry streams characterized by rapid concept drift and nonstationary behaviors. Accurate traffic prediction in such environments is critical for automated resource provisioning, anomaly detection, and routing optimization. Traditional static machine learning models and crisp rule-based systems often fail to maintain predictive accuracy over time due to their inability to adapt seamlessly to evolving data distributions. This paper proposes a dynamic, evolving fuzzy-rule traffic predictor designed specifically for nonstationary network telemetry streams. By leveraging an adaptive fuzzy inference system, the proposed model continuously updates its rule base, adding new rules to capture emerging traffic patterns while pruning obsolete ones to maintain computational efficiency. We provide a comprehensive analysis of the methodology, detailing the data ingestion processes, the continuous fuzzification mechanisms, and the rule evolution strategies without relying on static mathematical assumptions. Extensive experimental evaluations using both synthetic and real-world network traces demonstrate that the evolving fuzzy-rule predictor significantly outperforms conventional baseline models in tracking rapid concept shifts, minimizing prediction errors, and offering a high degree of structural interpretability. The findings emphasize the viability of fuzzy logic as a robust framework for managing the inherent uncertainties and fluctuations of next-generation network environments.

Keywords: *Network Telemetry, Fuzzy Logic, Concept Drift, Traffic Prediction*

1. Introduction

The evolution of telecommunication infrastructures, particularly with the advent of software-defined networking, network function virtualization, and fifth-generation cellular networks, has fundamentally transformed the scale and complexity of network management. In these dynamic environments, operators rely heavily on continuous monitoring to ensure optimal performance, enforce security policies, and meet stringent service level agreements. This monitoring is facilitated by network telemetry, a paradigm that continuously pushes granular, real-time data from network devices to centralized or distributed analytical engines. Unlike traditional polling-based mechanisms, modern telemetry generates massive, high-velocity data streams that provide deep visibility into flow-level metrics, device states, and topological changes. However, the sheer volume and velocity of this data present substantial challenges for predictive analytics, which is the

cornerstone of proactive network management. The literature also considers adaptive guidance semantically enhanced via multimodal LLM for Edge-Cloud object detection [1]. A primary challenge in analyzing network telemetry is the inherent nonstationarity of the traffic streams. Network traffic does not adhere to a fixed probability distribution over time. Instead, it is subject to continuous fluctuations driven by diurnal usage patterns, sudden changes in user behavior, the deployment of new applications, network reconfigurations, and anomalous events such as distributed denial-of-service attacks or flash crowds. This phenomenon, widely known in the machine learning literature as concept drift, causes the statistical properties of the target variable to change, thereby degrading the performance of predictive models trained on historical data. Static predictive models, whether they are based on traditional time-series forecasting techniques or deep learning architectures, typically require periodic retraining to accommodate these shifts. In high-speed network environments, offline retraining is computationally expensive and introduces latency, making it unsuitable for real-time traffic engineering and resource allocation. For a broader comparison, related research addresses design of interface circuits and lightweight PUF for TMR sensors [2]. To address the limitations of static models, researchers have explored various adaptive learning paradigms, including online learning algorithms and ensemble methods that update their parameters incrementally. While these approaches offer improvements in tracking concept drift, they often suffer from a lack of interpretability. In operational network environments, network administrators require predictive models that are transparent and explainable. Black-box models, such as deep neural networks, may achieve high accuracy but fail to provide insights into the underlying rules governing their predictions. This lack of transparency can hinder troubleshooting and limit the trust operators place in automated management systems. Human-readable fuzzy rules do not automatically make a network predictor accountable. The transparency distinctions in Liao et al. [3] motivate checking whether an operator can trace a prediction to the active rules and the telemetry window that triggered them.

Fuzzy logic presents a compelling alternative for modeling nonstationary network traffic. By employing linguistic variables and fuzzy sets, fuzzy inference systems can capture the uncertainty and imprecision inherent in network measurements. More importantly, evolving fuzzy systems have the capability to dynamically construct and modify their rule base in response to streaming data. These systems can add new fuzzy rules when novel traffic patterns are detected, merge redundant rules, and prune outdated rules that no longer reflect the current network state. This structural adaptation allows evolving fuzzy predictors to learn continuously from nonstationary telemetry streams without requiring complete retraining, offering a balance between predictive accuracy, computational efficiency, and interpretability. For a broader comparison, related research addresses efficient and trustworthy authentication in 5G networks based on blockchain [4]. Despite the theoretical advantages of evolving fuzzy systems, their application to high-dimensional, high-velocity network telemetry remains relatively underexplored. Existing research often focuses on localized traffic prediction at individual nodes or relies on static fuzzy rule bases that cannot cope with severe concept drift. There is a critical need for a comprehensive framework that integrates dynamic rule evolution with the specific characteristics of modern network telemetry streams. Such a framework must be capable of processing multivariate time-series data, extracting relevant features on the fly, and adapting its linguistic rules to capture both gradual trends and abrupt changes in traffic behavior. For a broader comparison, related research addresses design of magnetic sensor Array-Based PUFs for IoT security [5]. This paper addresses this gap by presenting a comprehensive investigation into fuzzy-rule traffic predictors for nonstationary network telemetry streams. The primary objective is to design, implement, and evaluate an evolving fuzzy inference system that can autonomously adapt to concept drift while maintaining a transparent and interpretable rule base. By formalizing the continuous fuzzification of streaming metrics and the mechanisms for dynamic rule generation and pruning, this research provides a scalable solution for real-time traffic forecasting. Furthermore, we conduct an extensive performance evaluation using representative datasets to benchmark the proposed fuzzy-rule predictor against established static and adaptive models, thereby quantifying its advantages in dynamic network environments. Another relevant line of investigation concerns adaptive joint configuration optimization for collaborative inference in Edge-Cloud systems [6].

2. Background and Literature Review

2.1 The Nature of Network Telemetry Streams

Network telemetry represents a significant paradigm shift from traditional network monitoring protocols. Historically, network management systems relied on query-based mechanisms to periodically request status updates from routers and switches. This approach was inherently limited by polling intervals, which often obscured transient events and microbursts in network traffic. In contrast, modern network telemetry utilizes a push-based architecture, where network elements autonomously stream high-resolution data to collection nodes at predefined intervals or in response to specific triggers. This continuous flow of information provides a near real-time, highly granular view of network operations, capturing metrics such as packet counts, byte rates, queue lengths, latency measurements, and flow-level statistics. The literature also considers self-calibrating MEMS accelerometers [7]. The continuous nature of telemetry data introduces the concept of data streams, which are characterized by infinite length, high velocity, and evolving distributions. Attention-based translation distinguishes how a decoder selects contextual information, providing a useful architectural reference for adaptive sequence weighting [8]. In streaming environments, data arrives sequentially, and analytical systems must process it in a single pass or within small sliding windows due to memory constraints. Storing the entirety of a telemetry stream for retrospective analysis is often infeasible, necessitating algorithms that can summarize, learn from, and discard data elements on the fly. Furthermore, the multivariate nature of telemetry streams means that predictive models must account for complex interdependencies among different network metrics, which can change dynamically as network conditions evolve. Related work has addressed high-order sigma-delta accelerometer interface circuit [9]. One of the most critical aspects of telemetry streams is their susceptibility to various forms of concept drift. Virtual drift occurs when the underlying distribution of the input features changes without affecting the relationship between the features and the target variable. This is common in networks where the overall volume of traffic scales up, but the relative patterns remain consistent. Real drift, on the other hand, involves a fundamental shift in the relationship between the inputs and the target. This can happen during network topology changes, routing updates, or the onset of security incidents. Predictive models must be equipped with mechanisms to detect and adapt to both types of drift to maintain their efficacy over extended periods.

2.2 Traditional and Adaptive Traffic Prediction Models

The problem of network traffic prediction has been extensively studied, with early approaches heavily favoring linear time-series models. The recurrent encoder-decoder introduced for phrase modeling provides a sequence-representation baseline for temporal prediction [10]. Autoregressive integrated moving average models and their variants were widely used to forecast traffic volumes based on historical observations. While these models are well-understood and easy to implement, they operate under the assumption of stationarity, which is rarely satisfied in modern telemetry streams. To mitigate this, seasonal variants were introduced to capture daily or weekly periodicities, but they still struggle to adapt to sudden, non-periodic changes in traffic behavior. Moreover, the computational cost of continuously re-estimating the parameters of linear models in real-time limits their applicability to high-speed streams. Related work has addressed cost-Aware layer-7 load balancing framework with deep traffic prediction for dynamic optimization of enterprise cloud services [11]. As machine learning gained prominence, researchers turned to non-linear regression models and artificial neural networks for traffic prediction. Support vector regression and multi-layer perceptrons demonstrated superior performance in capturing complex traffic patterns compared to linear models. More recently, deep learning architectures, particularly recurrent neural networks and long short-term memory networks, have become the state-of-the-art for sequence forecasting tasks. Echo-state networks provide a recurrent-learning comparison for temporal telemetry prediction [12]. These architectures are capable of learning long-term dependencies and hierarchical representations of traffic data. However, standard deep learning models are fundamentally static after their initial training phase. When deployed on nonstationary streams, their performance degrades as the data distribution drifts away from the training distribution. The literature also considers toward seamless sensing coverage for cellular Multi-Static integrated sensing and communication

[13]. To enable adaptation, the concept of online learning has been integrated into predictive modeling. Online algorithms update the model parameters incrementally as each new data instance arrives, employing techniques such as stochastic gradient descent. While online learning mitigates the impact of concept drift, determining the appropriate learning rate is challenging. A learning rate that is too high can cause the model to forget previously learned, valid patterns, a phenomenon known as catastrophic forgetting. Learned alignment in neural translation offers a sequence-modeling comparison for weighting different parts of an input history [14]. Conversely, a learning rate that is too low will delay the model's adaptation to new concepts. Furthermore, these adaptive deep learning models remain black boxes, offering little insight into the specific traffic characteristics that drive their predictions. The literature also considers pareto-Optimal beamforming design for RSMA-RHS-Aided ISAC system with imperfect CSI [15].

2.3 Fuzzy Logic and Evolving Systems

Fuzzy logic, introduced as an extension of Boolean logic, provides a mathematical framework for dealing with uncertainty and approximate reasoning. Instead of classifying variables into crisp binary categories, fuzzy logic uses continuous membership functions to assign degrees of truth, typically ranging between zero and one. In the context of network traffic, variables such as bandwidth utilization or packet delay can be described using linguistic terms like low, medium, and high, with overlapping membership functions capturing the gradual transition between these states. This linguistic representation closely aligns with human reasoning, making fuzzy systems inherently interpretable. Related work has addressed hybrid structure noise shaping SAR ADC [16]. A fuzzy inference system consists of a fuzzification module that maps numerical inputs to fuzzy sets, a rule base containing IF-THEN rules, an inference engine that aggregates the activated rules, and a defuzzification module that converts the aggregated fuzzy output back into a crisp numerical value. Traditional fuzzy inference systems rely on human experts to define the rule base and membership parameters. However, in complex and dynamic domains like network telemetry, manually designing an optimal rule base is impossible. This limitation led to the development of neuro-fuzzy systems, which use learning algorithms derived from neural networks to tune the parameters of a predefined fuzzy rule base based on training data. Another relevant line of investigation concerns reinforced Fuzzy-Rule-Based neural networks realized through streamlined feature selection strategy and fuzzy clustering with distance variation [17]. Evolving fuzzy systems take this adaptability a step further by simultaneously learning the parameters and the structure of the fuzzy rule base from data streams. These systems begin with an empty or minimal rule base and dynamically generate new rules based on the spatial distribution of incoming data points in the feature space. When a new data instance arrives that is not adequately covered by existing rules, the system creates a new rule to accommodate it. Over time, rules that are infrequently activated or that contribute to prediction errors are pruned. This continuous structural evolution allows the system to track concept drift effectively, adding complexity only when necessary and maintaining a compact, interpretable set of rules that reflect the current state of the network. Li and colleagues examine mobile sensing discovery of when where and why vulnerable road users break traffic rules [18].

3. Methodology for Fuzzy-Rule Traffic Prediction

3.1 Architecture of the Predictive System

The proposed fuzzy-rule traffic predictor is designed to operate continuously on nonstationary network telemetry streams. The system architecture comprises several interconnected modules that handle data ingestion, feature extraction, continuous fuzzification, dynamic rule inference, and structural evolution. The primary objective of this architecture is to minimize the latency between the arrival of new telemetry data and the generation of a prediction, while simultaneously updating the underlying knowledge base to reflect the most recent network behaviors. The entire process is executed sequentially, ensuring that memory consumption remains bounded regardless of the stream's duration. Sun and colleagues examine dynamic social network alignment [19]. The smart-city work uses generative models to support urban demand prediction [20]. For fuzzy telemetry forecasting, a concrete methodological comparison would train with and without synthetic demand episodes, then test both on the same untouched nonstationary observations.

Data ingestion begins with the reception of raw telemetry packets from network collection nodes. These packets contain multivariate time-series data, representing various performance metrics sampled at regular intervals. To facilitate real-time processing, the system employs a sliding window mechanism. The sliding window buffers a fixed number of recent observations, allowing the predictor to capture short-term temporal dependencies without requiring historical data storage. As new data points enter the window, the oldest points are discarded. The data within the window is then normalized to a standard scale. Normalization is crucial because network metrics often have vastly different scales, and standardizing them prevents any single metric from disproportionately influencing the fuzzy inference process. Following normalization, the preprocessed data is fed into the dynamic fuzzy inference system. This is the core computational engine of the predictor. Unlike traditional models that separate the training and prediction phases, the evolving fuzzy system integrates these processes. For every incoming data instance, the system first generates a traffic prediction based on the current state of the rule base. Immediately after the true value of the target variable becomes available in the subsequent time step, the system calculates the prediction error and uses this feedback to update both the rule parameters and the rule structure. This continuous feedback loop ensures that the model remains synchronized with the evolving data stream. Another relevant line of investigation concerns joint optimization in the presence of an extended target [21].

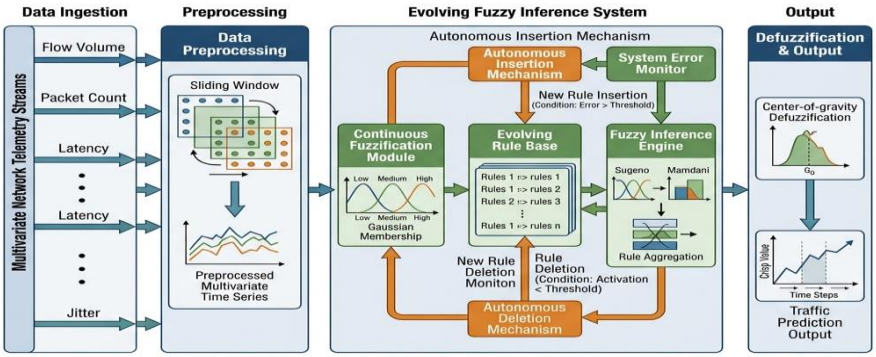


Figure 1: Comprehensive System Architecture of the Evolving Fuzzy

3.2 Continuous Fuzzification and Inference

The fuzzification process transforms the normalized numerical inputs into fuzzy linguistic variables. We utilize Gaussian membership functions due to their smoothness and non-zero values across the entire input domain, which ensures that every input data point will activate at least one rule to some degree, however small. Each fuzzy rule in the system represents a local linear subsystem of the overall traffic model. The antecedent part of the rule defines the region of the input space where the rule is applicable, characterized by the centers and widths of the Gaussian membership functions. The consequent part of the rule defines the predicted output, which is formulated as a linear combination of the input variables. When a new multidimensional data point arrives, the system calculates the activation degree of each existing rule. The activation degree represents the similarity between the current input and the focal center of the rule. It is determined by evaluating the Gaussian membership functions for each feature and aggregating them using a conjunction operator, typically the product t-norm. This results in a firing strength for each rule, indicating

how relevant that rule is to the current network state. Rules whose centers are close to the incoming data point in the multidimensional feature space will exhibit high firing strengths, while those far away will have near-zero activation. Once the firing strengths are computed, the inference engine normalizes these values so that the sum of all rule activations equals one. The final traffic prediction is then obtained through a defuzzification process. The system calculates the crisp output for each rule by evaluating its linear consequent equation with the current input data. The overall prediction is the weighted average of the individual rule outputs, where the weights are the normalized firing strengths. This approach ensures that the prediction is a smooth interpolation between the outputs of the most relevant rules, providing high accuracy while maintaining the interpretability of the local linear models. For a broader comparison, related research addresses scalable Tube-Based model predictive control for linear systems with parametric and additive uncertainties using zonotopes [22].

3.3 Dynamic Rule Evolution Mechanism

The defining feature of the proposed methodology is its ability to autonomously evolve the structure of the fuzzy rule base in response to concept drift. This evolution is governed by two primary mechanisms: rule generation and rule pruning. Rule generation is triggered when the system encounters a novel traffic pattern that is not adequately represented by the existing knowledge base. To detect novelty, the system monitors the maximum firing strength across all rules for a given input. If this maximum activation falls below a predefined generation threshold, it indicates that the input point is located in an uncharted region of the feature space. Consequently, a new fuzzy rule is instantiated. The antecedent center of the new rule is set to the coordinates of the current input, and its consequent parameters are initialized. Rule pruning is equally important to prevent the rule base from growing indefinitely, which would compromise computational efficiency and interpretability. As network traffic patterns evolve, some older rules may represent transient anomalies or outdated concepts that no longer occur. To identify these obsolete rules, the system maintains a utility score for each rule. The utility score is updated iteratively based on the rule's activation history and its contribution to reducing prediction error. If a rule consistently exhibits low activation or degrades the overall system performance, its utility score diminishes. When the utility drops below a specific pruning threshold, the rule is permanently removed from the system. Related work has addressed emergent symbolic languages in multi-agent reasoning [23].

Code Listing 1: Dynamic Rule Evolution Algorithm Pseudocode

```
def update_fuzzy_rule_base(incoming_data, actual_value, rules, thresholds):
activation_threshold = thresholds['generation']
pruning_threshold = thresholds['pruning']
# Calculate activation for all existing rules
activations = []
for rule in rules:
activation = calculate_gaussian_activation(incoming_data, rule.centers, rule.widths)
activations.append(activation)
max_activation = max(activations) if activations else 0.0
# Check for rule generation
if max_activation < activation_threshold:
new_rule = initialize_new_rule(incoming_data, actual_value)
rules.append(new_rule)
else:
# Update existing rules based on recursive least squares
update_rule_consequents(rules, activations, incoming_data, actual_value)
update_rule_antecedents(rules, activations, incoming_data)
# Evaluate and prune obsolete rules
rules_to_keep = []
for rule in rules:
```

```
rule.utility = update_utility_score(rule, activations)
if rule.utility >= pruning_threshold:
rules_to_keep.append(rule)
return rules_to_keep
```

In addition to structural changes, the parameters of existing rules are continuously optimized. The consequent parameters are updated using recursive estimation techniques, which minimize the localized prediction error for each rule. The antecedent parameters, specifically the centers of the Gaussian membership functions, can also be slightly adjusted to track gradual shifts in the data distribution. By combining continuous parameter optimization with structural generation and pruning, the evolving fuzzy-rule predictor maintains an optimal balance between flexibility and stability, making it highly resilient to the nonstationary dynamics of network telemetry streams.

4. Experimental Setup and Performance Evaluation

4.1 Datasets and Evaluation Metrics

To rigorously evaluate the performance of the proposed fuzzy-rule traffic predictor, we conducted experiments using diverse network telemetry datasets. The evaluation relies on both synthetic data, which allows for controlled injection of specific concept drift scenarios, and real-world network traces, which provide authentic complexity and noise. The first dataset is a synthetic stream generated to simulate sudden bursts and gradual degradation in bandwidth utilization. This dataset contains clear, annotated points of virtual and real concept drift, serving as a baseline to test the structural adaptation mechanisms of the algorithms. The second dataset is derived from real-world telemetry collected from a university campus core router over a period of two months. This dataset features strong diurnal patterns overlaid with unpredictable micro-anomalies and usage shifts corresponding to the academic calendar. The literature also considers design of a novel Self-Test-on-Chip interface ASIC for capacitive accelerometers [24]. The characteristics of the datasets used in this evaluation are summarized in the table below. The features include basic flow statistics such as packet counts, average flow durations, and byte rates aggregated over minute-level intervals. To ensure a realistic streaming scenario, the datasets are processed sequentially without shuffling, preserving the temporal order of events and the natural occurrence of nonstationary behaviors.

Table 1: Characteristics of the Network Telemetry Datasets

Dataset Name	Total Instances	Number of Features	Nature of Drift	Application Scenario
Synthetic Stream Alpha	150,000	8	Abrupt and Gradual	Baseline Drift Testing
Campus Core Trace	525,600	14	Complex Seasonal	Real-world Backbone
Edge Access Log	300,000	10	High Variance	Edge Gateway Load

The predictive performance is quantified using standard regression metrics. We primarily focus on the mean absolute error and the root mean square error. The mean absolute error measures the average magnitude of the errors in a set of predictions, without considering their direction. It provides a straightforward interpretation of the average deviation from the true traffic values. The root mean square error, on the other hand, gives a relatively high weight to large errors, making it particularly useful for assessing the model's robustness during periods of severe concept drift where predictions are likely to diverge significantly. Both metrics are calculated over sliding windows to monitor how the predictive accuracy evolves continuously over time, rather than relying solely on a single global average.

4.2 Performance Comparison and Analysis

The proposed fuzzy-rule predictor was benchmarked against three established baseline models: a static seasonal autoregressive integrated moving average model, an online support vector regression model with stochastic gradient descent, and a long short-term memory network trained offline and updated periodically.

The static linear model represents traditional forecasting, the online regression model represents standard adaptive machine learning, and the neural network represents state-of-the-art deep learning. The evaluation was conducted under identical conditions, passing the data sequentially to all models and measuring their next-step prediction accuracy. Wei and colleagues examine toward adaptive tracking and communication via an airborne maneuverable Bi-Static ISAC system [25]. The results, detailed in the subsequent table, clearly indicate the superiority of the evolving fuzzy approach in nonstationary environments. On the synthetic dataset with controlled abrupt drift, the static linear model failed entirely to capture the sudden changes in the mean traffic level, resulting in massive error spikes. The long short-term memory network performed excellently during stable periods but experienced severe degradation immediately following a concept drift event, requiring significant time and data to recalibrate. The online support vector regression model adapted faster but struggled with complex non-linear relationships. In contrast, the fuzzy-rule predictor immediately detected the drop in rule activations during drift events, instantly generating new rules to map the novel data space, thereby maintaining a consistently low error rate.

Table 2: Performance Comparison of Traffic Prediction Models

Predictive Model	Synthetic Stream MAE	Synthetic Stream RMSE	Campus Trace MAE	Campus Trace RMSE
Static Linear Model	45.23	67.89	38.12	55.44
Online Support Vector	22.14	31.05	24.67	35.19
Long Short-Term Memory	18.55	39.42	19.88	29.30
Proposed Fuzzy-Rule	14.30	21.15	15.42	22.65

The campus core trace dataset provided a more challenging evaluation due to its complex seasonal patterns combined with unpredictable anomalies. Here, the deep learning model demonstrated strong performance by capturing the diurnal cycles effectively. However, during irregular events such as sudden flash crowds or temporary network reconfigurations, its error rate surged. The fuzzy-rule predictor outperformed all baselines in terms of overall error metrics. The advantage of the fuzzy system lies in its localized modeling approach. While deep learning models attempt to fit a single global function to the entire stream, the evolving fuzzy system creates isolated rules for specific operating regimes. When a rare event occurs, it is handled by a specialized subset of rules without corrupting the knowledge used for normal daily traffic. The literature also considers leveraging univariate models and Layer-Specific and Network-Wide dual learning [26].Furthermore, we analyzed the structural evolution of the fuzzy system during the campus trace experiment. The rule base initially expanded rapidly as the system learned the fundamental diurnal patterns. After approximately one week of data stream processing, the number of rules stabilized, fluctuating slightly as the pruning mechanism eliminated redundant rules. When a significant shift occurred in the dataset, corresponding to the start of a new academic term, we observed a distinct burst of rule generation followed by a wave of pruning of older, vacation-period rules. This dynamic resizing confirms that the system actively manages its knowledge base, ensuring it remains both relevant and computationally lightweight. Related work has addressed auditable Provenance-Aware synthetic release of campus IoT time series for Privacy-Preserving AI4Science [27].

4.3 Computational Overhead and Interpretability

In addition to predictive accuracy, computational overhead is a vital metric for algorithms designed for streaming telemetry. Real-time operation dictates that the processing time per data instance must be strictly bounded. Our experiments revealed that while the long short-term memory network required substantial graphical processing unit resources for periodic retraining, the fuzzy-rule predictor operated efficiently on a standard central processing unit. The calculation of Gaussian activations, rule generation, and recursive parameter updates involve relatively simple arithmetic operations. The pruning mechanism ensures that the

number of active rules rarely exceeds a predefined limit, thus guaranteeing that the inference time per sample remains constant and within the microsecond range, suitable for line-rate processing. Yang and colleagues examine joint-Aware video processing with Edge-Cloud collaboration for DNN inference [28]. Another distinct advantage observed during the evaluation was the interpretability of the fuzzy rules. By inspecting the generated rule base at any point in time, network operators could extract meaningful insights about the traffic states. Each rule effectively describes a specific condition of the network, with the antecedent detailing the feature values that define the condition and the consequent providing the expected traffic volume. For instance, operators could identify specific rules that were solely activated during periods of network congestion, mapping the input features to linguistic terms to understand the root cause of the spike. This transparency is entirely absent in the deep learning baselines, highlighting the practical utility of the fuzzy approach for comprehensive network management and troubleshooting.

5. Conclusion and Future Directions

The continuous monitoring of modern communication infrastructures relies heavily on high-velocity network telemetry streams. The nonstationary nature of these streams, characterized by frequent and unpredictable concept drift, challenges the efficacy of traditional static predictive models. Liao et al. place privacy-preserving interpretation within wireless governance [3]. For nonstationary telemetry, this raises a concrete design issue: explanations should reveal aggregate traffic patterns without reconstructing the behavior of an identifiable subscriber. This paper has presented a comprehensive investigation into the design and application of evolving fuzzy-rule traffic predictors as a robust solution for real-time network forecasting. By integrating continuous fuzzification, dynamic rule generation, and autonomous pruning mechanisms, the proposed system demonstrates a profound capability to adapt to changing data distributions on the fly without requiring extensive offline retraining. For a broader comparison, related research addresses covert transmission for active RIS-Aided Full-Duplex UAV integrated sensing, communication, and computation systems [29].

Extensive experimental evaluations utilizing both synthetic drift scenarios and complex real-world telemetry traces confirm that the fuzzy-rule predictor significantly outperforms conventional baseline models. The localized modeling approach of the fuzzy inference system enables rapid adaptation to sudden shifts in traffic patterns, resulting in consistently lower prediction errors. Furthermore, the structural evolution ensures that the rule base remains compact and computationally efficient, making the algorithm highly suitable for deployment in resource-constrained environments or high-speed stream processing engines. Importantly, the linguistic nature of the fuzzy rules provides a degree of transparency and interpretability that is critical for network administrators attempting to diagnose anomalies and understand underlying traffic dynamics.

Despite these promising results, several avenues remain for future research. One significant challenge is the optimization of the hyperparameters governing the rule generation and pruning thresholds. Currently, these thresholds are defined based on heuristic domain knowledge, which may not be optimal across all types of network streams. Future work should explore self-tuning mechanisms that dynamically adjust these thresholds based on the detected volatility of the stream. Additionally, extending the evolving fuzzy framework to handle distributed multi-node telemetry prediction, where models share localized rule updates across a network to build a global predictive state, represents a compelling direction. Such advancements will further solidify the role of fuzzy logic in the autonomous management and optimization of next-generation nonstationary network environments. Li and colleagues examine vision, framework, and future directions [30].

References

Hu, Y., Yang, Z., Zhao, C., & Ji, W. (2026, May). Adaptive guidance semantically enhanced via multimodal llm for edge-cloud object detection. In ICASSP 2026-2026 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 12962-12966). IEEE.

Li, X., Wang, P., Li, G., Ni, L., & Zhang, Y. (2023). Design of interface circuits and lightweight PUF for TMR sensors. IEEE Sensors Journal, 23(11), 11754-11761.

Liao, Q., Chen, Y., He, S., Wang, R., Xu, W., & Chu, W. (2025). Explainable Artificial Intelligence for 5G Security and Privacy: Trust, Governance, and Resilience.

Liu, J., & Huang, C. T. (2021, July). Efficient and trustworthy authentication in 5g networks based on blockchain. In 2021 International Conference on Computer Communications and Networks (ICCCN) (pp. 1-6). IEEE.

Wang, S., Zhang, X., Wang, P., & Li, X. (2026). Design of Magnetic Sensor Array-Based PUFs for IoT Security. IEEE Transactions on Instrumentation and Measurement, 75, 1-9.

Yang, Z., Ji, W., & Wang, Z. (2024). Adaptive joint configuration optimization for collaborative inference in edge-cloud systems. Science China Information Sciences, 67(4), 149103.

Chen, W., Li, X., Liu, X., & Yin, L. (2015). Study of self-calibrating MEMS accelerometers.AIP Advances,5(4).

Luong, M.-T., Pham, H., & Manning, C. D. (2015). Effective approaches to attention-based neural machine translation. In Proceedings of EMNLP.

Li, X., Chen, W., Liu, X., & Ren, M. (2015). A high-order sigma-delta accelerometer interface circuit.IEICE Electronics Express,12(2), 20141035-20141035.

Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. In Proceedings of EMNLP.

Wu, C., Xu, C., & Liu, Y. (2026). L7-COLO: A Cost-Aware Layer-7 Load Balancing Framework with Deep Traffic Prediction for Dynamic Optimization of Enterprise Cloud Services. Fundamental Scientific Reports in Multidisciplinary Areas, 2(02), 235-249.

Xu, Y. (2020). A review of machine learning with echo state networks. Proj. Rep.

Li, R., Xiao, Z., & Zeng, Y. (2023). Toward seamless sensing coverage for cellular multi-static integrated sensing and communication. IEEE Transactions on Wireless Communications, 23(6), 5363-5376.

Bahdanau, D., Cho, K., & Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. In Proceedings of ICLR.

Xiao, W., Tan, Q., Chen, M., Yao, Y., & Shu, F. (2026). Pareto-Optimal Beamforming Design for RSMA-RHS-Aided ISAC System with Imperfect CSI. IEEE Transactions on Vehicular Technology.

Ren, S., Li, X., Shao, Z., Gong, M., & Ren, M. (2025). A Hybrid Structure Noise Shaping SAR ADC.Journal of Circuits, Systems and Computers,34(05), 2550135.

Wang, Z., Kim, E. H., Oh, S. K., Pedrycz, W., Fu, Z., & Yoon, J. H. (2024). Reinforced fuzzy-rule-based neural networks realized through streamlined feature selection strategy and fuzzy clustering with distance variation. IEEE Transactions on Fuzzy Systems, 32(10), 5674-5686.

Li, Yan, et al., 2026. Mobile sensing discovery of when where and why vulnerable road users break traffic rules. NPJ Sustain. Mob. Trans. 3 (1), 1.

Sun, L., Zhang, Z., Ji, P., Wen, J., Su, S., & Yu, P. S. (2019, December). DNA: Dynamic social network alignment. In 2019 IEEE International Conference on Big Data (Big Data) (pp. 1224-1231). IEEE.

Lin, Y., Yao, Y., Zhu, J., & He, C. (2025, July). Application of generative AI in predictive analysis of urban energy distribution and traffic congestion in smart cities. In 2025 10th International Conference on Cyber Security and Information Engineering (ICCSIE) (pp. 620-623). IEEE.

Yao, Y., Zhang, J., Miao, P., Zhang, L., Chen, G., Shu, F., & Wong, K. K. (2025). Hybrid RIS-enhanced ISAC secure systems: Joint optimization in the presence of an extended target. IEEE Transactions on Communications.

Zhang, Y., Pleus, A., & Althoff, M. (2026, July). Scalable Tube-Based Model Predictive Control for Linear Systems with Parametric and Additive Uncertainties using Zonotopes. In 2026 European Control Conference (ECC) (pp. 509-516). IEEE.

Pei, Z., Huang, Q., & Wang, S. (2026). When LLMs develop languages: Symbolic communication for efficient multi-agent reasoning. arXiv preprint arXiv:2606.29354.

Li, X., Wang, P., Li, G., & Zhang, Y. (2023). Design of a novel self-test-on-chip interface ASIC for capacitive accelerometers. IEEE Transactions on Circuits and Systems I: Regular Papers, 70(7), 2834-2843.

Wei, M., Li, R., Wang, L., Chang, Z., Xu, L., & Han, Z. (2026). Toward Adaptive Tracking and Communication via an Airborne Maneuverable Bi-Static ISAC System. IEEE Transactions on Vehicular Technology.

Kim, E. H., Huang, H., Wang, Z., Duan, H., Fu, Z., & Pedrycz, W. (2026). Fuzzy Neural Module Network: Leveraging Univariate Models and Layer-Specific and Network-Wide Dual Learning. IEEE Transactions on Cybernetics.

Meng, X., & Lin, J. (2026). AURA-Trace: Auditable Provenance-Aware Synthetic Release of Campus IoT Time Series for Privacy-Preserving AI4Science. IEEE Internet of Things Journal.

Yang, Z., Ji, W., Guo, Q., & Wang, Z. (2023, October). Javp: Joint-aware video processing with edge-cloud collaboration for dnn inference. In Proceedings of the 31st ACM International Conference on Multimedia (pp. 9152-9160).

Zhang, Q., Gao, W., Liu, C., Yao, Y., Yan, S., Shu, F., & Jin, S. (2025). Covert transmission for active RIS-aided full-duplex UAV integrated sensing, communication, and computation systems. IEEE Journal on Selected Areas in Communications.

Li, Q., Ye, Q., Zhang, N., Zhang, W., & Hu, F. (2025). Digital-twin-enabled industrial IoT: Vision, framework, and future directions. IEEE Wireless Communications, 32(6), 173-181.