

BiLCNet : BiLSTM-Conformer Network for Encrypted Traffic Classification with 5G SA Physical Channel Records

Ke Ma

*SJTU Paris Elite Institute of Technology
Shanghai Jiao Tong University
Shanghai, China
vanessa0717@sjtu.edu.cn*

Jialiang Lu

*SJTU Paris Elite Institute of Technology
Shanghai Jiao Tong University
Shanghai, China
jjialiang.lu@sjtu.edu.cn*

Philippe Martins

*Telecom Paris
Institut Polytechnique de Paris
Palaiseau, France
martins@telecom-paris.fr*

Abstract—Accurate and efficient traffic classification is vital for wireless network management under encrypted payloads and dynamic application behavior, where traditional methods such as port-based methods and deep packet inspection (DPI) are increasingly inadequate. This work explores the feasibility of using physical channel data collected from the air interface of 5G Standalone (SA) networks for traffic sensing. We design a preprocessing pipeline to convert raw channel records into structured representations with customized feature engineering. To jointly capture temporal dependencies and both local and global structural patterns in these records, we propose a novel hybrid architecture: BiLSTM-Conformer Network (BiLCNet), which integrates the sequential modeling capability of Bidirectional Long Short-Term Memory networks (BiLSTM) with the spatial feature extraction strength of Conformer blocks. Evaluated on a noise-limited 5G SA dataset, BiLCNet achieves 94.23% accuracy, outperforming conventional machine learning and deep learning baselines. Ablation results further verify the contribution of each module. We also evaluate domain generalization across varying channel conditions, demonstrating robustness across traffic categories and environmental variations.

Index Terms—encrypted traffic classification, physical channel data, BiLSTM-Conformer Network, domain generalization testing.

I. INTRODUCTION

Traffic classification is a fundamental task in modern mobile networks, enabling effective resource allocation, content-aware billing, access control, and anomaly detection. Its performance directly affects quality of service (QoS), network security, and management efficiency [1]. However, the widespread adoption of encryption in services such as e-commerce, social media, and video streaming has made accurate traffic identification increasingly challenging [2]. Traditional methods such as port-based approaches and deep packet inspection (DPI) are no longer sufficient, highlighting the need for advanced learning-based techniques capable of handling encrypted and diverse traffic in real time.

A. Traditional Traffic Classification Techniques

Port-based identification is the oldest traffic classification method that maps known Transmission Control Protocol/User Datagram Protocol (TCP/UDP) ports to services. While simple

and efficient, its reliability has declined due to dynamic ports, tunneling, and Network Address Port Translation (NAPT) [3].

Another traditional approach is DPI, which analyzes packet headers and payloads using pattern matching to identify applications. While it achieves high accuracy for unencrypted traffic, its high computational cost and poor performance under encryption significantly limit its practical applicability [4].

B. Learning-based Traffic Classification Techniques

Learning-based methods, mainly including machine learning (ML) and deep learning (DL) algorithms, have emerged to address the limitations of traditional approaches under encryption. Representative learning-based methods are reviewed in the following, with key approaches and their challenges summarized in Table I.

Supervised learning methods such as K-Nearest Neighbors (KNN) [5], Naive Bayes [6], and Support Vector Machines (SVM) have shown promise in traffic classification but face key limitations: data distribution sensitivity, dependence on labeled data, and limited real-time use. In contrast, unsupervised approaches like K-means [7] and Expectation-Maximization (EM) [8] remove the need for labeled data but often suffer from unstable performance and rigid clustering structures, leading to reduced accuracy for complex traffic patterns. In addition, semi-supervised learning [9] has also been widely explored, though it typically requires more computational resources than both supervised and unsupervised methods.

In recent years, deep learning has become a mainstream approach for encrypted traffic classification, with Convolutional Neural Networks (CNNs) [10], Graph Neural Networks (GNNs) [11], and Recurrent Neural Networks (RNNs) [12] widely used to extract temporal features. More recently, Transformer-based models have shown strong capabilities in traffic analysis due to their attention mechanisms and ability to capture long-range dependencies. There have also been various attempts to combine different architectures, such as integrating CNN with LSTM [13] or fusing CNN and Transformer models [14]. However, robust generalization across diverse channel conditions and sparsity in records remain key challenges for cross-layer sensing in 5G networks.

TABLE I
COMPARISON OF LEARNING-BASED TECHNIQUES IN TRAFFIC CLASSIFICATION

Technique	Algorithm	Input Features	Main Challenge
Supervised Learning	KNN [5] Naive Bayes [6]	Network traffic statistical features TCP/UDP flow statistical features	Poor distinction between streaming and bulk flows Strong dependence on dataset distribution
Unsupervised Learning	K-means [7] EM [8]	TCP flow packet-level features Network traffic statistical features	High sensitivity to initial cluster center selection Low separability with similar feature distributions
Deep Learning	CNN [10] GNN [11] RNN [12] CNN + LSTM [13] CNN + Transformer [14]	Raw radio time series Network traffic graph features Time-domain amplitude and phase Network traffic statistical features Network traffic statistical features	Reduced classification accuracy at low SNR Structural limitations with increasing depth Reduced classification accuracy at low SNR Poor performance on imbalanced datasets Limited generalization to zero-day traffic patterns

To address the above limitations, we propose BiLCNet for encrypted traffic classification. Compared with our prior 5G NSA study [15] and LightGBM-based 5G SA study [16], this work advances this line from NSA feasibility and feature-based ML to sequence-aware deep modeling of 5G SA physical-channel records, with an emphasis on cross-gain generalization. The principal contributions of this work include:

- We collect 5G SA physical channel data under diverse channel conditions and design a preprocessing pipeline to construct informative channel feature matrices.
- We propose BiLCNet, which integrates Bidirectional LSTM for temporal modeling, CNN for local feature extraction, and Transformer architecture for global attention to improve classification performance.
- Our model targets robust accuracy and out-of-domain generalization across diverse traffic patterns and unseen channel conditions, supporting traffic-aware dimensioning in offline network planning.

II. PRELIMINARIES

A. 5G Mobile Communication Network Architecture

The 5G mobile communication network represents a significant upgrade from Fourth Generation Long-Term Evolution (4G LTE), offering higher speed, lower latency, and greater capacity. As shown in Fig. 1, its architecture consists of three core components: the User Equipment (UE), such as smartphones and IoT devices; the Radio Access Network (RAN), which manages wireless connections and allocates radio resources; and the 5G Core Network (5GC), responsible

for overall control and data routing. Wireless links connect the UE and RAN, while the RAN and core network communicate through control and user plane interfaces.

The simulation platform used to generate the dataset in this study is built upon this standard 5G architecture, specifically adopting the Standalone (SA) deployment mode. Unlike the Non-Standalone (NSA) architecture, which integrates 5G radio access with the existing 4G LTE infrastructure, SA operates independently, employing the 5GC and gNodeB (gNB) as its core and access components, respectively. Selecting the SA deployment for our experimental platform results in a simplified network stack and more faithfully represents native 5G application scenarios.

B. Physical Channels of the 5G Wireless Air Interface

In 5G networks, wireless communication between the UE and gNB involves logical, transport, and physical channels. This study focuses on physical channel data from the 5G air interface, which can reflect service-level variations without packet decryption, thereby preserving privacy and reducing computational overhead. Since this information is directly linked to radio resource usage and upper-layer traffic behavior, it provides useful cues for accurate and timely encrypted traffic classification.

Specifically, 5G physical channels operate within the physical (PHY) layer and are responsible for the actual transmission of signals over radio resources. There are six standardized physical channels in 5G: the Physical Uplink/Downlink Control Channel (PUCCH/PDCCH), Physical Uplink/Downlink Shared Channel (PUSCH/PDSCH), Physical Broadcast Channel (PBCH), and Physical Random Access Channel (PRACH). Their transmission content is summarized in Table II.

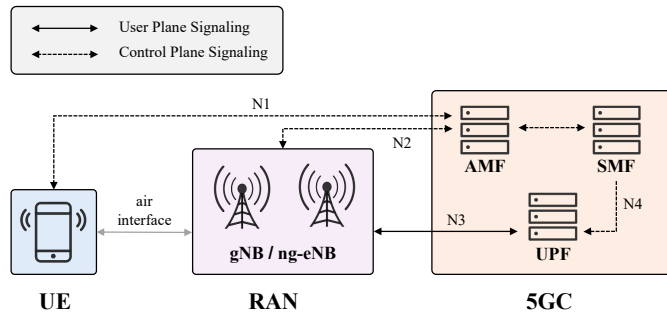


Fig. 1. 5G Mobile Communication Network Architecture

TABLE II
SUMMARY OF 5G PHYSICAL CHANNELS

Channel	Direction	Transmission Content
PDCCH	Downlink	Physical Layer Control information
PDSCH	Downlink	User data
PBCH	Downlink	System information for access
PUCCH	Uplink	Physical Layer Control information
PUSCH	Uplink	User data
PRACH	Uplink	Signals for random access attempts

III. METHODOLOGY

A. Data Collection

Our 5G physical channel dataset was collected in a Faraday cage at the INFRES¹ laboratory to eliminate external electromagnetic interference. The experimental platform simulates a realistic 5G communication environment by integrating the functions of both the RAN and the 5GC on a single computer running the Amarisoft² 5G protocol stack. This computer is connected to a software-defined radio (SDR) transceiver, which handles the transmission and reception of wireless signals. On the user side, a commercial 5G smartphone serves as the UE, establishing a wireless air interface link with the SDR. This setup allows us to simulate various user behaviors on the UE while capturing the corresponding physical-layer log data through Amarisoft's data acquisition software.

Overall, we collected 5G physical channel data across four distinct user behavior patterns and eleven different channel conditions. The behavior patterns considered were (1) voice calls, (2) video meetings, (3) large file uploads, and (4) large file downloads. To simulate varying wireless channel qualities, the transmission gain of the access network was adjusted from 64 dB to 84 dB in increments of 2 dB. Higher transmission gain values correspond to better channel conditions.

B. Data Preprocessing

The raw dataset consists of log files containing physical channel feature data. We design a lightweight data preprocessing pipeline aligned with the 5G SA frame structure to transform these logs into structured, trainable data representations, as illustrated in Fig. 2.

5G employs orthogonal frequency-division multiple access (OFDMA) to divide bandwidth into interference-free subcarriers for fine-grained resource allocation. In 5G SA systems, the physical layer adopts a fixed radio frame length of 10 ms, with each frame comprising 10 subframes of 1 ms. For numerology

$\mu = 1$, each subframe is further divided into 2 slots, and each time-frequency unit corresponds to a resource block (RB).

Based on this frame structure, the preprocessing pipeline operates as follows. In a 10 ms frame-based observation window, for each subframe, we extract features from four key physical channels: PUCCH, PUSCH, PDSCH, and PDCCH, and concatenate them into a fixed-length Subframe Feature Vector. Zero-padding is applied if a channel is missing. The feature vectors of all 10 subframes are then stacked to form a Channel Feature Matrix, providing a compact and consistent input representation for downstream learning tasks.

The dataset includes measurements from four physical channels, but many features are sparse. To improve computational performance, we retain over 60 informative features selected based on a feature relevance analysis reported in our prior work [15]. The key variables are summarized in the table III.

TABLE III
PHYSICAL MEANING OF KEY SELECTED FEATURES

Category	Feature	Physical Meaning
User Data Stream	<i>mcs</i>	Modulation and Coding Scheme
	<i>harq</i>	Hybrid Automatic Repeat Request
	<i>tb_len</i>	Length of Transport Block
	<i>prb</i>	Physical Resource Block Allocation
Air Interface Environment	<i>snr</i>	Signal-to-Noise Ratio at Receiver
	<i>epr</i>	Energy per Resource Element
	<i>cce</i>	Control Channel Element

Following the extraction of the aforementioned features, additional feature engineering [16] techniques are employed to improve the model's generalization capability. In particular, a range of derived statistical descriptors is computed to reveal implicit relationships within the data.

1) *Cumulative Error Rate*: The cumulative uplink/downlink error rate (ERR) is calculated based on Hybrid Automatic Repeat Request (HARQ) feedback as follows:

$$ERR = 1 - \frac{N_{\text{succ}}}{N_{\text{total}}} \quad (1)$$

where N_{succ} is the number of successful HARQ transmissions and N_{total} the total HARQ attempts. A higher ERR means less reliable traffic type needing stronger error correction, while a lower ERR indicates more stable transmission.

2) *PDSCH Efficiency*: The efficiency of PDSCH is defined as the ratio between the total transmitted transport block size and the total number of Physical Resource Blocks (PRBs) allocated:

$$EFF_{\text{PDSCH}} = \frac{\sum TB_{\text{PDSCH}}}{\sum PRB_{\text{PDSCH}}} \quad (2)$$

A higher EFF_{PDSCH} indicates more efficient utilization of downlink resources, typically associated with real-time or latency-sensitive services.

3) *Modulation Variability Index*: The Modulation Variability Index (MVI) quantifies the fluctuation in modulation schemes

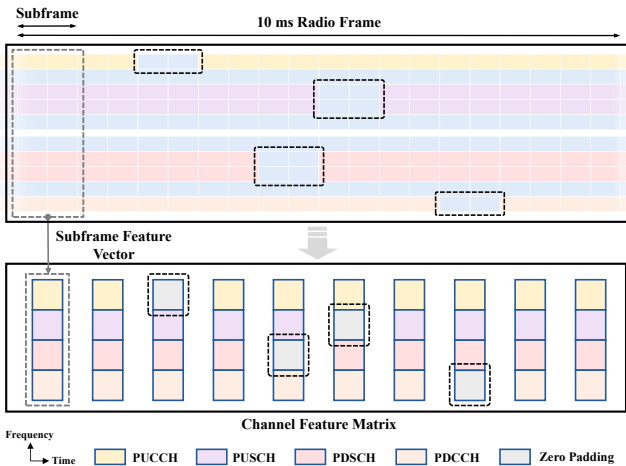


Fig. 2. Data Preprocessing of 5G SA Physical Channel Features ($\mu = 1$)

¹Department of Computer Science and Networks, Télécom Paris

²Amarisoft official website: <https://www.amarisoft.com/>

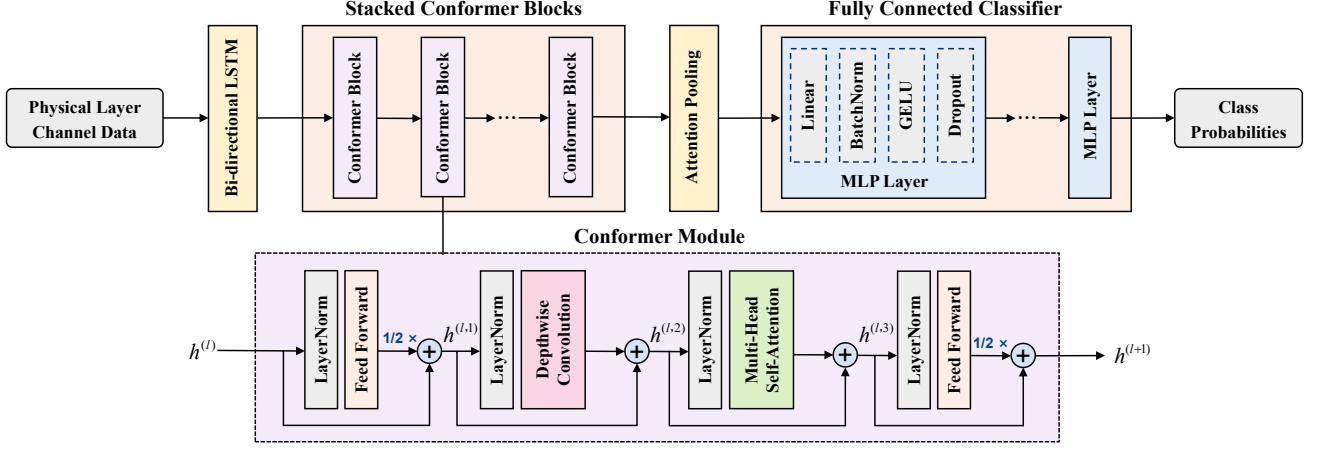


Fig. 3. Architecture of BiLCNet

within a given observation window. It is defined as the coefficient of variation of the modulation order:

$$\text{MVI} = \frac{\sigma_{\text{mod}}}{\mu_{\text{mod}}} \quad (3)$$

where σ_{mod} and μ_{mod} denote the standard deviation and mean of modulation order values, respectively. A lower MVI indicates stable modulation, while a higher MVI reflects greater variability due to adaptive modulation in dynamic conditions.

C. BiLCNet : BiLSTM-Conformer Network

Through data preprocessing, we construct frame-structured channel feature matrices from physical-layer logs. Each subframe is encoded as a feature vector, with ten vectors stacked to form a matrix. This representation encodes both temporal dependencies across subframes and spatial correlations within each subframe. Therefore, to effectively exploit these characteristics, we design BiLCNet, a hybrid model illustrated in Fig. 3. It combines a bidirectional LSTM for temporal modeling, while the Conformer blocks jointly learn spatial correlations and temporal dependencies through convolution and self-attention.

Bidirectional LSTM Layer To model temporal dependencies along the sequence, the input feature matrix $\mathbf{X}_t \in \mathbb{R}^{T \times D}$, where T is the number of subframes and D is the feature dimension of each subframe, is first fed into a bidirectional LSTM (BiLSTM) with L layers and hidden dimension H . The BiLSTM processes sequential data in both forward and backward directions to capture contextual information. At each time step t , the hidden states $\mathbf{H}_t \in \mathbb{R}^{n \times 2H}$ are computed as:

$$\mathbf{H}_t = [\text{LSTM}^{(f)}(\mathbf{X}_t, \vec{\mathbf{H}}_{t-1}); \text{LSTM}^{(b)}(\mathbf{X}_t, \overleftarrow{\mathbf{H}}_{t+1})] \quad (4)$$

where n denotes the batch size, $\text{LSTM}^{(f)}$ and $\text{LSTM}^{(b)}$ denote the forward and backward LSTM operations respectively, and $[\cdot; \cdot]$ represents concatenation. The final output $\mathbf{O}_t \in \mathbb{R}^{T \times 2H}$ is obtained through a linear transformation of $\mathbf{H}_t$.

BiLSTMs inherently capture sequential dependencies in both forward and backward directions, naturally modeling positional relationships within sequences. This allows them to effectively replace positional embeddings in Transformer-based models, as the BiLSTM's recurrent nature already preserves order information through its hidden states.

Conformer Block To capture both global and local inter-frame spatial dependencies, a stack of Conformer blocks is applied after the BiLSTM layer. The Conformer [17] is a convolution-augmented Transformer that integrates multi-head self-attention with convolutional modules to model long-range dependencies while preserving sensitivity to local patterns. Let $h^{(l)} \in \mathbb{R}^{T \times 2H}$ be the input to the l -th Conformer layer. The output $h^{(l+1)}$ is computed through the following residual computations:

$$h^{(l,1)} = h^{(l)} + \frac{1}{2} \cdot \text{FFN}(\text{LayerNorm}(h^{(l)})) \quad (5)$$

$$h^{(l,2)} = h^{(l,1)} + \text{Conv}(\text{LayerNorm}(h^{(l,1)})) \quad (6)$$

$$h^{(l,3)} = h^{(l,2)} + \text{MHSA}(\text{LayerNorm}(h^{(l,2)})) \quad (7)$$

$$h^{(l+1)} = h^{(l,3)} + \frac{1}{2} \cdot \text{FFN}(\text{LayerNorm}(h^{(l,3)})) \quad (8)$$

Here, MHSA denotes the Multi-Head Self-Attention mechanism defined as:

$$\text{MHSA}(Q, K, V) = \text{Concat}(\text{head}_1, \dots, \text{head}_h)W^O \quad (9)$$

where each attention head is computed as:

$$\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V) \quad (10)$$

where Q , K , and V denote the query, key, and value matrices, obtained via linear projections of the input. W_i^Q , W_i^K , and W_i^V are learnable projection matrices for the i -th head, and W^O is the output projection matrix.

Each module in this Conformer block uses residual connections and layer normalization for stable optimization, and convolution-attention fusion enriches feature representations by combining local and global information.

Attention Pooling Following the Conformer blocks, attention pooling is applied to aggregate features across the sequence. Given the output $h^{(l+1)} \in \mathbb{R}^{T \times 2H}$, the attention weights $\mathbf{w}$ are computed as:

$$\mathbf{w} = \text{Softmax}(\text{Linear}(h^{(l+1)})) \quad (11)$$

and the final pooled feature representation h_p is calculated as:

$$h_p = \sum_{t=1}^T \mathbf{w}_t \cdot h_t^{(l+1)} \quad (12)$$

This pooling mechanism enables the model to focus on the most informative parts of the sequence dynamically.

Fully Connected Classifier Finally, the pooled feature vector is passed through a fully connected network with linear layers, batch normalization, GELU activation, and dropout. Batch normalization stabilizes training, GELU introduces non-linearity, and dropout helps prevent overfitting, enabling the model to generalize better.

Loss Function and Optimizer To guide the model training, we use the cross-entropy loss function to quantify the difference between predicted and true labels. For optimization, we employ the Adam with Weight Decay (AdamW) algorithm, which combines momentum and weight decay to enhance convergence and reduce overfitting.

IV. EXPERIMENTAL EVALUATION

In this section, the learning frameworks are implemented in Python 3.12.5 using PyTorch 2.4.1. Experiments are conducted on NVIDIA GeForce RTX 4070 Laptop GPU (CUDA 12.1), using approximately 870,000 samples collected from four physical channels across four traffic categories.

A. Multi-Scenario Learning

Multi-scenario learning enables models to learn from diverse channel conditions. In our experiment, the model is trained on physical channel data collected under 11 wireless channel quality levels. To reduce data leakage, we preserve temporal order when splitting the dataset instead of random shuffling, since temporally adjacent samples are often highly correlated. Full hyperparameters are provided in the GitHub repository linked at the end of the paper.

We adopt accuracy (AC), precision (PR), recall (RC), and F1-score (F1) as evaluation metrics to assess the classification performance. Table IV summarizes the results of BiLCNet across different traffic types, where AC is not shown per

TABLE IV
MULTI-SCENARIO LEARNING PERFORMANCE OF BiLCNET

Index	User Behavior	AC	PR	RC	F1
(1)	Call	—	74.85%	87.16%	80.54%
(2)	Meeting	—	90.14%	90.01%	90.08%
(3)	Upload	—	98.86%	97.08%	97.96%
(4)	Download	—	98.24%	95.18%	96.69%
Overall metrics		94.23%	90.52%	92.36%	91.32%

TABLE V
BASELINE COMPARISON AND BiLCNET ABLATION
IN MULTI-SCENARIO LEARNING

Group	Models	AC	PR	RC	F1
ML	LR	91.32%	87.50%	89.30%	88.25%
	CART	91.32%	87.03%	87.66%	87.32%
	RF	92.71%	88.74%	90.12%	90.12%
	LGBM	92.93%	89.24%	90.75%	89.91%
DL	LSTM	93.40%	89.84%	91.40%	90.53%
	Transformer	93.45%	90.11%	90.86%	90.45%
Ablation	w/o Conformer	93.31%	89.31%	91.47%	90.12%
	w/o BiLSTM	93.56%	90.24%	90.89%	90.52%
	w/o Conv	93.54%	89.86%	91.26%	90.44%
	BiLCNet (Full)	94.23%	90.52%	92.36%	91.32%

class because it is defined over the entire test set. The model performs particularly well on *Upload* and *Download* traffic, which can be attributed to the distinct and stable feature patterns of these traffic types. In contrast, the results on *Call* and *Meeting* are relatively lower, possibly due to their lower occurrence or higher feature similarity.

To further validate BiLCNet, Table V compares it with ML/DL baselines and ablation variants under the multi-scenario setting. BiLCNet (Full) achieves the best overall performance, surpassing both ML baselines and standalone DL models. The ablation results show that removing the Conformer, BiLSTM, or convolutional component consistently reduces performance, confirming the contribution of each module. Although the improvements appear marginal in percentage terms, they represent meaningful progress given the large dataset scale. All ablation variants are trained under identical settings, indicating that combining temporal modeling, attention, and local feature extraction is beneficial for capturing complex traffic patterns.

Fig. 4 evaluates 40-epoch training trajectories of the three



Fig. 4. Deep Learning Model Comparison: Metric Evolution Across Epochs

deep learning architectures under multi-scenario conditions, with the vertical axis magnified for clearer comparison. The results demonstrate that the Transformer exhibits significant instability during training, while the traditional LSTM shows better stability but with an evident performance ceiling. In contrast, BiLCNet maintains superior accuracy across both training and validation phases, while also achieving the fastest convergence speed during initial training epochs.

B. Leave-One-Domain-Out evaluation

Leave-One-Domain-Out (LODO) evaluation measures a model's ability to generalize from seen to unseen channel conditions. In this study, we partition scenarios based on the transmission gain of wireless access networks, using data from each of the 11 gain settings as the test set while training the model on the remaining data. As shown in Fig. 5, each bar represents the model's performance (min-max over repeated runs) on a test set from an unseen gain setting, with the vertical axis scaled from 0.5 to 1 to highlight the variations.

The experimental results show that BiLCNet achieves an average accuracy of 83.4% across the tested transmission gain spectrum, demonstrating robust overall performance. Additionally, The model exhibits a characteristic "low-high-low" performance trend, with optimal accuracy observed in the 66-80 dB range, indicating strong domain generalization ability within this interval. However, the performance degrades at both gain extremes. This suggests that the model's LODO evaluation requires training samples from diverse channel conditions, where both high and low-quality channel knowledge are essential for adapting to new environments. These findings highlight opportunities to enhance model robustness across wider signal strength variations for real-world deployment.

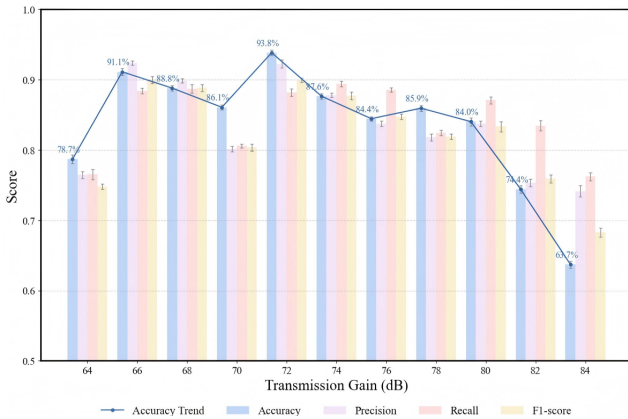


Fig. 5. LODO Domain Generalization Performance of BiLCNet

V. CONCLUSION

This paper shows that, by leveraging 5G SA physical channel records, encrypted traffic can be classified without payload inspection by converting short PHY log sequences into structured channel feature matrices. On a large-scale multi-scenario dataset, BiLCNet achieves 94.23% accuracy and shows consistent gains over ML/DL baselines, while the LODO results

indicate a certain level of cross-gain generalization. Although the controlled Faraday cage setup limits environmental variability, these results suggest that PHY-derived features remain a promising direction for intelligent traffic management, with broader real-world validation left for future work.

Codes and instructions of this work are available at <https://github.com/make78457/5G-SA-Traffic-Classification>.

REFERENCES

- [1] A. Azab, M. Khasawneh, S. Alrabaa, K.-K. R. Choo, and M. Sarsour, "Network traffic classification: Techniques, datasets, and challenges," *Digital Communications and Networks*, vol. 10, no. 3, pp. 676–692, 2024.
- [2] P. Wang, X. Chen, F. Ye, and Z. Sun, "A survey of techniques for mobile service encrypted traffic classification using deep learning," *Ieee Access*, vol. 7, pp. 54 024–54 033, 2019.
- [3] A. Madhukar and C. Williamson, "A longitudinal study of p2p traffic classification," in *14th IEEE international symposium on modeling, analysis, and simulation*. IEEE, 2006, pp. 179–188.
- [4] M. Finsterbusch, C. Richter, E. Rocha, J.-A. Muller, and K. Hanssgen, "A survey of payload-based traffic classification approaches," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 2, pp. 1135–1156, 2013.
- [5] M. Roughan, S. Sen, O. Spatscheck, and N. Duffield, "Class-of-service mapping for qos: a statistical signature-based approach to ip traffic classification," in *Proceedings of the 4th ACM SIGCOMM conference on Internet measurement*, 2004, pp. 135–148.
- [6] A. W. Moore and D. Zuev, "Internet traffic classification using bayesian analysis techniques," in *Proceedings of the 2005 ACM SIGMETRICS international conference on Measurement and modeling of computer systems*, 2005, pp. 50–60.
- [7] L. Bernalle, R. Teixeira, I. Akodkenou, A. Soule, and K. Salamatin, "Traffic classification on the fly," *ACM SIGCOMM Computer Communication Review*, vol. 36, no. 2, pp. 23–26, 2006.
- [8] S. Zander, T. Nguyen, and G. Armitage, "Automated traffic classification and application identification using machine learning," in *The IEEE Conference on Local Computer Networks 30th Anniversary (LCN'05) I*. IEEE, 2005, pp. 250–257.
- [9] T. Glennan, C. Leckie, and S. M. Erfani, "Improved classification of known and unknown network traffic flows using semi-supervised machine learning," in *Australasian conference on information security and privacy*. Springer, 2016, pp. 493–501.
- [10] T. J. O'Shea, J. Corgan, and T. C. Clancy, "Convolutional radio modulation recognition networks," in *International conference on engineering applications of neural networks*. Springer, 2016, pp. 213–226.
- [11] B. Pang, Y. Fu, S. Ren, Y. Wang, Q. Liao, and Y. Jia, "Cggn: traffic classification with graph neural network," *arXiv preprint arXiv:2110.09726*, 2021.
- [12] S. Rajendran, W. Meert, D. Giustiniano, V. Lenders, and S. Pollin, "Deep learning models for wireless signal classification with distributed low-cost spectrum sensors," *IEEE Transactions on Cognitive Communications and Networking*, vol. 4, no. 3, pp. 433–445, 2018.
- [13] W. Wang, Y. Sheng, J. Wang, X. Zeng, X. Ye, Y. Huang, and M. Zhu, "Fast-ids: Learning hierarchical spatial-temporal features using deep neural networks to improve intrusion detection," *IEEE access*, vol. 6, pp. 1792–1806, 2017.
- [14] Z. Liu, Y. Xie, Y. Luo, Y. Wang, and X. Ji, "Transec-net: A transformer-based model for encrypted traffic classification," *Applied Sciences*, vol. 15, no. 6, p. 2977, 2025.
- [15] X. Fei, P. Martins, and J. Lu, "Real-time traffic classification for 5g nsa encrypted data flows with physical channel records," in *2023 IEEE 98th Vehicular Technology Conference (VTC2023-Fall)*. IEEE, 2023, pp. 1–6.
- [16] B. Aphayavong, X. Fei, J. Lu, and P. Martins, "Optimizing lgbm for multi-classification of 5g sa traffic," in *2024 IEEE 100th Vehicular Technology Conference (VTC2024-Fall)*. IEEE, 2024, pp. 1–7.
- [17] A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu, and R. Pang, "Conformer: Convolution-augmented Transformer for Speech Recognition," in *Interspeech 2020*, 2020, pp. 5036–5040.