Lightweight LLM-based End-to-End CSI Prediction for MIMO-OFDM Systems

Xiongli RUI, Rui CHEN, Xiaoyan ZHAO, Jie YANG, Jinxiang CHEN

School of Communication and Artificial Intelligence, Nanjing Institute of Technology, Nanjing, 211167, China

{ruixiongli, chenrui, xiaoyanzhao, yangjie}@njit.edu.cn, jinxiangchen2025@163.com

Submitted March 24, 2026 / Accepted July 19, 2026 / Online first July 31, 2026

Abstract. *Accurate channel state information (CSI) is essential for multiple-input multiple-output orthogonal frequency division multiplexing (MIMO-OFDM) systems, yet fast time-varying channels pose significant prediction challenges. Traditional approaches fail under high mobility, while deep learning methods rely heavily on large labeled datasets, limiting generalization with scarce training data. Although large language models (LLMs) show promise, their massive parameter count hinders deployment on resource-constrained edge devices. This paper proposes a lightweight, end-to-end CSI prediction framework built upon a general LLM. A time-frequency dual-domain feature extraction module captures subcarrier correlations and temporal dynamics from historical CSI, overcoming single-domain limitations. The end-to-end design maps historical CSI directly to future states, avoiding error propagation inherent in explicit channel estimation. Parameter efficiency is achieved through low-rank adaptation (LoRA) combined with knowledge distillation from a pre-trained LLM, enabling effective few-shot learning at low computational cost. Simulations demonstrate that the proposed scheme delivers robust prediction accuracy across diverse mobility scenarios, maintains strong performance under limited training data, and exhibits zero-shot cross-scenario generalization, significantly outperforming both conventional and deep learning baselines in TDD and FDD modes.*

Keywords

Channel prediction, multiple-input multiple-output (MIMO), orthogonal frequency division multiplexing (OFDM), large language model (LLM), knowledge distillation (KD), low-rank adaptation (LoRA)

1. Introduction

Massive multiple-input multiple-output (MIMO) technology and orthogonal frequency division multiplexing (OFDM) are core foundational technologies for 5G and 6G wireless communication systems. The combination of MIMO and OFDM forms the dominant MIMO-OFDM transmission framework that significantly improves system spectral efficiency and transmission reliability [1]. Channel state information (CSI) refers to a set of critical parameters in wireless communication systems, which characterize the effects of the channel on signal transmission. Specifically, these parameters capture the channel fading, path loss, and interference characteristics between the transmitter and receiver. They further serve as the foundation for dynamically adjusting beamforming weights and optimizing resource allocation strategies, such as subcarrier and power allocation. Theoretically, the performance of massive MIMO is governed by the accuracy of the CSI [2], [3]. Only with accurate CSI can a system fully exploit the advantages of MIMO in spatial multiplexing and diversity gain, and thereby ensure high-speed, low-latency communication quality [4].

However, acquiring accurate CSI remains a persistent challenge in modern wireless communication systems [5]. First, pilot contamination introduces inherent inaccuracies to initial CSI estimation [6]. As pilot signals are often reused among users or cells, mutual interference contaminates channel estimation and reduces CSI fidelity. Second, feedback induced latency and imperfections in frequency division duplex (FDD) systems exacerbate CSI degradation [7]. In FDD architectures, user equipment (UE) must quantize and transmit estimated downlink CSI to the base station (BS) via low-rate uplink feedback channels. This process inevitably introduces signaling overhead and delay, making the CSI obtained at the BS outdated, especially under fast time-varying channels. Third, UE mobility emerges as a critical bottleneck, particularly in high-speed scenarios [8]. Rapid UE movement shortens channel coherence time, leading to fast-varying channel characteristics that render previously estimated CSI obsolete and unusable for key techniques like multi-user precoding. When combined with feedback latency, this mobility induced channel variation further widens the gap between the BS's stored CSI and the actual channel state [9], [10].

To address these issues and obtain high quality CSI in dynamic environments, conventional approaches primarily rely on three strategies, increasing the density of pilot signals [11], enhancing channel estimation algorithms [12], [13], or optimizing channel interpolation schemes. However, these methods face inherent limitations. For example, the number of pilot signals must increase rapidly with UE speed to track fast CSI variations. This reduces the effective data rate and increases energy consumption, making it impractical for

DOI: 10.13164/re.2026.0392

high-mobility scenarios [14]. An alternative approach to balance CSI quality and overhead is channel prediction, leveraging temporal correlations between historical CSIs and future CSIs [15].

Existing channel prediction methods face inherent limitations. Model-based approaches heavily rely on idealized propagation assumptions, leading to degraded accuracy in real-world multipath environments. Deep learning-based methods typically treat CSI prediction as pure time-series regression, overlooking space-time-frequency coupling characteristics, which limits generalization under high-mobility scenarios. Meanwhile, emerging large language models (LLM) based prediction solutions, although exhibiting strong cross-domain adaptation, suffer from massive parameter sizes and prohibitive inference costs, rendering them unsuitable for edge deployment [16], [17]. A comprehensive review and comparison of existing approaches is provided in Sec. 2.

Motivated by this, we propose a lightweight LLM-based channel prediction method. Specifically, based on the Qwen2.5-Math-1.5B model [18], we introduce a series of adaptive optimizations and targeted enhancements tailored to the unique requirements of CSI prediction, including cross-domain transfer learning, parameter fine-tuning, and the integration of data alignment techniques. These improvements collectively yield an efficient and accurate CSI prediction scheme, which is well-suited for MIMO-OFDM systems. The main works of this paper are summarized as follows:

- An end-to-end lightweight LLM based CSI prediction framework for high-mobility MIMO-OFDM systems is proposed. The end-to-end architecture directly maps historical CSI to future states, circumventing cascaded errors from explicit channel estimation in conventional multi-stage workflows.
- A time-frequency dual-domain feature extraction module is developed to simultaneously capture frequency-domain subcarrier correlations and time-domain temporal evolution characteristics from raw CSI sequences. This overcomes the inherent limitations of single-domain feature extraction and enhances the characterization of multi-dimensional CSI dynamics.
- We enable effective few-shot learning by combining low-rank adaptation (LoRA) with knowledge distillation (KD) from pre-trained LLMs. This significantly reduces computational complexity and trainable parameters while retaining strong generalization performance, making the framework suitable for resource-constrained devices.

The remainder of this paper is organized as follows. Section 2 reviews related works on channel prediction. Section 3 presents the system model for MIMO-OFDM channel prediction. Section 4 describes the proposed lightweight LLM-based CSI prediction framework, including the time-frequency dual-domain preprocessing module, KD strategy, and LoRA-driven fine-tuning mechanism. Section 5 provides the experimental setup and analyzes the simulation results. Finally, Section 6 concludes the paper.

2. Related Works

Current channel prediction approaches can be broadly categorized into three types: model-based methods, deep learning-based methods, and physics-informed hybrid methods. Model-based prediction techniques primarily include parametric models [19] and autoregressive (AR) approaches. However, their effectiveness heavily relies on predefined propagation assumptions. Owing to the complex multipath characteristics of real wireless channels, mismatches between assumed and actual channels limit prediction accuracy [20]. In contrast, deep learning-based methods, including deep neural network (DNN) [21], recurrent neural network (RNN) [22], [23], gated recurrent unit (GRU) [24], long short-term memory (LSTM) [25], [26], convolutional neural network (CNN) [27], generative adversarial network (GAN) [28], [29] as well as conventional Transformer [30], [31], achieve better fitting capability than model-driven solutions. Nevertheless, most existing deep learning methods regard CSI prediction as pure time-series regression and neglect the inherent space-time-frequency coupling of path loss, multipath and shadow fading, resulting in poor generalization under high-speed and time-varying scenarios. This limits their generalization and practical flexibility. Some improved deep learning variants have been proposed to alleviate these defects. In [32], comprehensive experimental comparisons are conducted among typical neural networks including CNN, RNN and Transformer under various moving speeds to reveal their practical pros and cons. Different from common deterministic prediction, the conformal Bayes filtering architecture in [33] combines quantile regression and Bayesian filtering to implement uncertainty evaluation for CSI prediction results. In addition, KD is utilized to shrink network size for CSI feedback tasks in [34]. Even with these optimizations, physics-informed hybrid networks and the above improved models still cannot fully resolve the drawbacks of unreasonable feature fusion and poor environmental adaptability.

LLMs exhibit strong modeling and generalization ability, showing great potential in time-series analysis. Recently, several works have begun to explore LLM-based channel prediction [35]. For single time-series CSI input, reference [36] adapts the native next token prediction of generative pre-trained transformer-2 (GPT-2) to realize variable length input, while reference [37] further constructs cross-modal semantic alignment between CSI features and text embedding, and adopts attention-based KD to obtain lightweight student model with outstanding limited-data performance. To exploit propagation environment information, sensing-aided LLM reference [38] fuses integrated sensing and communication (ISAC) echo and communication CSI. Besides, reference [39] adopts a cloud-edge collaborative architecture and introduces image and light detection and ranging (LiDAR) information to mine environmental features for

Category	Reference	Solution	Base Model	Lightweight	Few-Shot	Generalization	Use Case
Model-driven	[19], [20]	Prony + PFT parametric modeling	AR / Prony	✓	✗	Weak	High-speed terrestrial MIMO
DL-RNN	[22]–[26]	Recurrent time-series regression	LSTM / GRU / RNN	✓	✗	Weak	Ordinary cellular links
DL-CNN / GAN	[27], [28], [29]	2D feature + adversarial generation	RNN / LSTM / GRU	✓	✗	Weak	CSI repair & MIMO fitting
DL-Transformer	[30], [31]	Self-attention long-sequence modeling	Vanilla Transformer	✗	✗	Strong	Single-scene high-dynamic MIMO
LLM-Vanilla	[35], [36], [39], [41]	CSI token + LoRA fine-tuning	GPT-style LLM	✓	Strong	Moderate	General 6G multi-scenario
LLM-KD	[37]	Modal alignment + knowledge distill	LLaMA-3-1B + LoRA	✗	Partial	Moderate	Common multi-user MIMO
LLM-Custom	[38], [40]	GPT2+ConvLSTM cross-fusion	GPT-2	✗	Partial	Moderate	Sensing-aided / OTFS satellite
Proposed Framework	This work	Dual-domain input + LoRA fine-tuning	DeepSeek-R1-Distill-Qwen-1.5B	✓	Strong	Strong	High-mobility MIMO-OFDM

Tab. 1. Comparison of model-driven, DL-based and LLM-based channel prediction methods.

channel prediction and beamforming, together with a proposed multimodal LLM framework. For low Earth orbit (LEO) satellite links, the scheme in [40] compresses channel dimensions and adopts LORA fine-tuning, and builds a prediction framework tailored for systems combining orthogonal time frequency space (OTFS) and fluid antenna system (FAS). The model in [41] leverages multimodal data and zero-shot learning for universal channel prediction, yet it suffers from excessive parameters and insufficient specialized dataset support. For a clearer illustration of the pros and cons of mainstream prediction solutions including generalization ability, lightweight property and practical deployment conditions, a comprehensive classification summary is provided in Tab. 1.

Benefiting from pre-trained knowledge, LLMs can quickly adapt to diverse channel environments, overcoming the heavy reliance on scenario-specific data of traditional methods. However, LLMs usually have massive parameters, resulting in low inference speed and high computation cost. For example, the 175B-parameter GPT-3 requires at least 350GB graphics processing unit (GPU) memory under half-precision floating-point (FP16) [42], making it difficult to deploy on resource-limited edge devices such as mobile phones and embedded systems. For next-generation wireless communications that require accurate, efficient, and edge-deployable CSI prediction, developing lightweight LLM-based channel prediction models with balanced performance and resource cost has become an important research direction.

3. System Model

As illustrated in Fig. 1(a), we consider a massive MIMO-OFDM downlink system deployed for autonomous vehicle communications inside logistics industrial parks, operating with N_c orthogonal subcarriers. The BS employs a uniform planar array (UPA) in the *x*-*z* plane, composed of $N_t = N_x \times N_z$ transmit elements, with respective antenna spacings d_x and d_z, as shown in Fig. 1(b). The UPA is configured and capable of resolving the direction of departure (DoD), which is characterized by the azimuth angle $\theta \in [-\pi, \pi)$ and the elevation angle $\varphi \in [-\pi/2, \pi/2)$. The UE is equipped with N_r receiving antennas, which form a uniform linear array (ULA) and enable the resolution of the direction of arrival (DOA). To optimize downlink transmission strategies, the BS conducts channel prediction in a high-mobility scenario, where the UE moves rapidly, and data is transmitted over time-varying fading channels. With pilot symbols interleaved in the transmitter's data stream, we use CSIs from P historical time steps to forecast the CSIs for the following L future time steps.

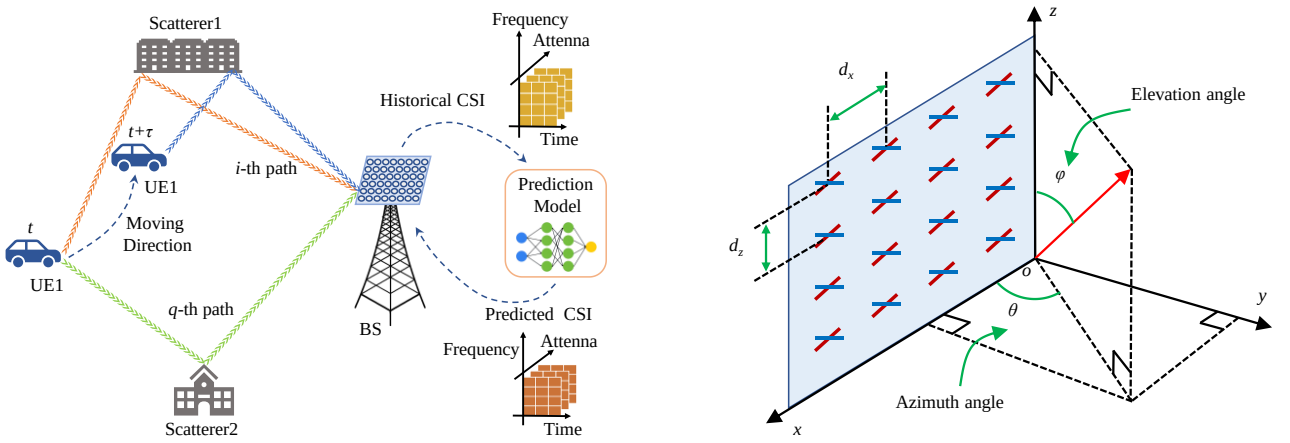
3.1 Channel Model

Adopting the classic clustering based multipath channel model [43], we assume there are U paths in a multipath propagation environment, which are divided into N clusters using a clustering algorithm. Each cluster contains M paths with similar delay or angular spread (satisfying $U = N \times M$). The downlink CSI between the BS and the UE at the i-th time and k-th frequency can be modeled as the linear superposition of multipath components [44]:

$$\mathbf{H}_{i,k} = \sum_{p=1}^{U} a_p \mathbf{s}_t\left(\varphi_p, \theta_p\right) \mathbf{s}_r\left(\rho_p\right) \mathrm{e}^{\mathrm{j}\left(2\pi\left(\omega_p i\Delta t - \tau_p k\Delta f\right) + \Phi_p\right)} \tag{1}$$

where $\mathbf{H}_{i,k} \in \mathbb{C}^{N_x N_z \times 1}$, $\mathrm{j} = \sqrt{-1}$. Δt and Δf are the symbol duration and subcarrier spacing, respectively. For the p-th propagation path, a_p, τ_p, and ϕ_p denote the complex path gain, propagation delay and random phase, while φ_p, θ_p and ρ_p represent the elevation angle of departure (EAD), azimuth angle of departure (AOD) and azimuth angle of arrival (AOA). ω_p stands for the Doppler frequency shift of the p-th path and is formulated as $\omega_p = (vf \cos\phi_p) / c$, where v is the instantaneous velocity of the UE at time t, c is the velocity of light, and ϕ_p denotes the angle between the electromagnetic wave propagation direction and the UE's velocity vector. The three-dimensional spatial steering vector of the p-th path, $\mathbf{s}_t(\varphi_p, \theta_p) \in \mathbb{C}^{N_x N_z \times 1}$, characterizes the elevation and azimuth angles of the path and is expressed as [45]:

$$\mathbf{s}_t\left(\varphi_p, \theta_p\right) = \mathrm{vec}\left(\mathbf{v}_x\left(\varphi_p, \theta_p\right) \otimes \mathbf{v}_z\left(\varphi_p, \theta_p\right)^{\mathrm{T}}\right) \tag{2}$$



(a) MIMO-OFDM system with moving UE

(b) The architecture of UPA.

Fig. 1. An illustration of system model.

where $\mathbf{v}_z(\varphi_p, \theta_p) \in \mathbb{C}^{N_z \times 1}$ and $\mathbf{v}_x(\varphi_p, \theta_p) \in \mathbb{C}^{N_x \times 1}$ are the vertical and horizontal steering vector, respectively. vec(·) denotes vectorization operation, ⊗ denotes the Kronecker product, and the superscript T denotes the transpose. Let $u = 2\pi / \lambda$ where λ is the carrier wavelength, The steering vectors $\mathbf{v}_z(\varphi_p, \theta_p)$ and $\mathbf{v}_x(\varphi_p, \theta_p)$ can be derived as:

$$\mathbf{v}_{\mathrm{x}}\left(\varphi_p, \theta_p\right) = [1, \mathrm{e}^{\mathrm{j}ud_x \cos\varphi_p \cos\theta_p}, \cdots, \mathrm{e}^{\mathrm{j}u(N_x-1)d_x \cos\varphi_p \cos\theta_p}]^{\mathrm{T}}, \quad (3)$$

$$\mathbf{v}_{\mathrm{z}}\left(\varphi_p, \theta_p\right) = [1, \mathrm{e}^{\mathrm{j}ud_z \cos\varphi_p \sin\theta_p}, \cdots, \mathrm{e}^{\mathrm{j}u(N_z-1)d_z \cos\varphi_p \sin\theta_p}]^{\mathrm{T}}. \quad (4)$$

$\mathbf{s}_{\mathrm{r}}(\rho_p)$ is the receive array response vector and can be expressed as [45]:

$$\mathbf{s}_{\mathrm{r}}\left(\rho_p\right) = [1, \mathrm{e}^{\mathrm{j}ud_{\mathrm{r}} \sin\rho_p}, \cdots, \mathrm{e}^{\mathrm{j}ud_{\mathrm{r}}(N_{\mathrm{r}}-1)\sin\rho_p}]^{\mathrm{T}} \quad (5)$$

where d_{r} denotes the spacing between the ULA antenna elements of UE.

The channel response matrix at the i-th time and k-th subcarrier for a MIMO-OFDM system with N_{t} transmit antennas and N_{r} receive antennas can be expressed as:

$$\mathbf{H}_{i,k} = \begin{bmatrix} h_{i,k,1,1} & h_{i,k,2,1} & \cdots & h_{i,k,N_{\mathrm{t}},1} \\ h_{i,k,1,2} & h_{i,k,2,2} & \cdots & h_{i,k,N_{\mathrm{t}},2} \\ \vdots & \vdots & \ddots & \vdots \\ h_{i,k,1,N_{\mathrm{r}}} & h_{i,k,2,N_{\mathrm{r}}} & \cdots & h_{i,k,N_{\mathrm{t}},N_{\mathrm{r}}} \end{bmatrix} \quad (6)$$

where $h_{i,k,m,n}$ denotes the multipath channel response between the m-th transmit antenna and n-th receive antenna. Even for paths within the same cluster, their parameters can differ slightly; each scatterer may exhibit distinct reflection angles for its corresponding paths, while it is also common for different paths to share one or more identical parameters. Derived from (1) and (6), the integrated CSI across all subcarriers at the i-th time step can be expressed as:

$$\mathbf{H}_i = \left[\mathbf{H}_{i,1}^{\mathrm{H}}, \mathbf{H}_{i,2}^{\mathrm{H}}, \ldots, \mathbf{H}_{i,N_{\mathrm{c}}}^{\mathrm{H}}\right]^{\mathrm{T}} \quad (7)$$

where the superscript symbol H denotes the conjugate transpose, and N_{c} is the number of subcarriers. $\mathbf{H}_i$ is an $N_{\mathrm{r}} \times N_{\mathrm{t}} \times N_{\mathrm{c}}$ CSI tensor that aggregates the $N_{\mathrm{r}} \times N_{\mathrm{t}}$ channel response matrices $\mathbf{H}_{i,k}$ (k = 1, 2,…, N_{c}) across all subcarriers at the i-th time step.

3.2 Signal Model

Considering transmit precoding at the BS, the received downlink signal of the UE at the k-th subcarrier and the i-th time can be derived as:

$$\mathbf{y}_{i,k} = \mathbf{H}_{i,k}^{\mathrm{H}} \mathbf{C}_{i,k} \mathbf{x}_{i,k} + \mathbf{n}_{i,k} \quad (8)$$

where $\mathbf{y}_{i,k} \in \mathbb{C}^{N_{\mathrm{r}} \times 1}$, $\mathbf{x}_{i,k} \in \mathbb{C}^{N_{\mathrm{str}} \times 1}$ and $\mathbf{n}_{i,k} \in \mathbb{C}^{N_{\mathrm{r}} \times 1}$ are the received signal vector, transmitted data symbol vector and the additive white Gaussian noise (AWGN) with noise power σ_{n}^2, respectively. $\mathbf{H}_{i,k} \in \mathbb{C}^{N_{\mathrm{r}} \times N_{\mathrm{t}}}$ and $\mathbf{C}_{i,k} \in \mathbb{C}^{N_{\mathrm{t}} \times N_{\mathrm{str}}}$ are the channel and transmit precoding matrixes, where $N_{\mathrm{str}} \leq (N_{\mathrm{t}}, N_{\mathrm{r}})$ denotes the number of data streams.

3.3 Problem Formulation

In the time division duplexing (TDD) based MIMO-OFDM systems, uplink and downlink share the same frequency band, leading to approximate channel reciprocity. This allows downlink CSI to be directly predicted from uplink CSI. For FDD systems, channel reciprocity does not hold due to frequency-division duplexing. Instead, downlink CSI estimated at the UE is fed back to the BS via the uplink, albeit with associated delay and signaling overhead. From a carrier perspective, both TDD uplink CSI and FDD downlink CSI represent carrier channel state information. In both scenarios, historical and future CSI are correlated. A well-designed learning model can extract spatiotemporal correlation patterns from historical CSI to establish a mapping Ψ: past CSI→future CSI. Thus, the downlink channel prediction problem for both TDD and FDD can be described as learning a functional mapping from P historical CSI time steps to L future CSI time steps, expressed as:

$$\left\{\mathbf{H}_{i+1}, \cdots, \mathbf{H}_{i+L}\right\} = \Psi\left(\left\{\mathbf{H}_{i-P+1}, \cdots, \mathbf{H}_i\right\}\right) \quad (9)$$

where Ψ(·) denote the mapping function, i is the current time step. P and L are the numbers of historical and predicted future time steps. Determining an effective nonlinear mapping function poses a significant challenge due to the inherent complexity and dynamic spatiotemporal variations of wireless environments. Traditional model-based designs often

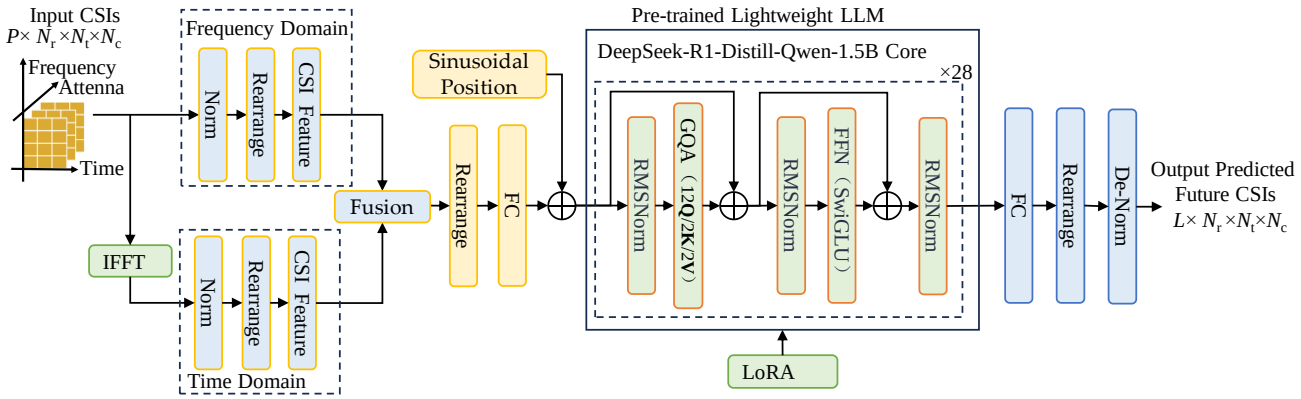


Fig. 2. An illustration of CSI prediction model.

struggle to achieve this, especially when the number of channel parameters becomes sufficiently large in rich, time-varying transmission scenarios.

In this paper, we propose a novel approach based on a lightweight pre-trained language model to address the downlink CSI prediction problem. The objective is to leverage its powerful capabilities in sequence modeling and long-range dependency capture to achieve higher prediction accuracy and superior generalization compared to traditional deep learning networks. The problem is formulated as:

$$\min_{\mathbf{\Lambda}} \mathrm{NMSE} = \mathbb{E}\left[\frac{\sum_{l=1}^{L}\left\|\hat{\mathbf{H}}_{i+l}-\mathbf{H}_{i+l}\right\|_{\mathrm{F}}^{2}}{\sum_{l=1}^{L}\left\|\mathbf{H}_{i+l}\right\|_{\mathrm{F}}^{2}}\right] \tag{10}$$

$$\text{s.t.}\left(\hat{\mathbf{H}}_{i+j},\ldots,\hat{\mathbf{H}}_{i+L}\right)=g_{\mathbf{\Lambda}}\left(\mathbf{H}_{i-P+1},\ldots,\mathbf{H}_{i}\right)$$

where $\hat{\mathbf{H}}_{i+l}$ and $\mathbf{H}_{i+l}$ represent the predicted and actual downlink CSI at time $i+l$, respectively, while $\mathbf{H}_{i-P+1}$, …, $\mathbf{H}_i$ denote the actual historical CSI over P time steps. $\|\ \|_{\mathrm{F}}^2$ is the squared Frobenius norm, which comprehensively measures the deviation between predicted and true CSI across temporal, frequency, and spatial dimensions, making it a standard evaluation criterion in channel prediction. $g_{\mathbf{\Lambda}}(\cdot)$ is the concrete implementation of the true mapping function $\Psi(\cdot)$, with $\mathbf{\Lambda}$ being the hyperparameter space of the lightweight LLM. Training parameters are tuned during the training process to minimize the normalized mean squared error (NMSE) metric, which quantifies the proximity between $g_{\mathbf{\Lambda}}(\cdot)$ and $\Psi(\cdot)$.

4. LLM Based CSI Prediction Model

LLMs are general purpose models for natural language processing (NLP) and cross-modal tasks, and lightweight domain specific variants are typically adapted from pre-trained base models via two mainstream approaches: fine-tuning techniques (e.g., LoRA) to embed domain knowledge, and KD to reduce model parameters while retaining teacher-model performance for edge deployment. Figure 2 illustrates the lightweight LLM based CSI prediction model, optimized using KD and LoRA fine-tuning. Aligned with standard LLM architectures, the model comprises three core components: input preprocessing and embedding, a pre-trained LLM backbone, and an output projection module.

4.1 Input Preprocessing and Embedding

As shown in Fig. 2, the prediction model takes the frequency domain historical CSI sequence $\mathbf{X} \in \mathbb{C}^{P\times N_{\mathrm{r}}\times N_{\mathrm{t}}\times N_{\mathrm{c}}}$ as input. To satisfy the real-valued input requirement of the LLM, the real and imaginary parts of the complex frequency domain CSI over P time steps are split into two real valued dimensions, forming a real-valued tensor $\mathbf{X}_{\mathrm{f}}[l] \in \mathbb{R}^{2\times N_{\mathrm{c}}\times N_{\mathrm{r}}\times N_{\mathrm{t}}}$ with $l \in \{i-P+1, i-P+2, \ldots, i\}$, where the dimension 2 corresponds to the real and imaginary parts. Since the time domain contains delay information that helps capture channel characteristics, we transform the frequency domain CSI into the time domain via inverse fast Fourier transform (IFFT) to jointly exploit both domains. For IFFT efficiency, if N_{c} is not an integer power of 2, the frequency domain sequence is zero-padded to length $2^{\lceil \log_2 N_{\mathrm{c}} \rceil}$, where $\lceil\cdot\rceil$ denotes the ceiling function. The resulting real valued time domain representation is $\mathbf{X}_{\mathrm{d}}[l] \in \mathbb{R}^{P\times N_{\mathrm{r}}\times N_{\mathrm{t}}\times N_{\mathrm{c}}}$, consistent with the frequency domain tensor dimension.

Subsequently, both $\mathbf{X}_{\mathrm{f}}[l]$ and $\mathbf{X}_{\mathrm{d}}[l]$ are flattened to fit the sequence input format of the LLM: for each time step, the four dimensions (transmit antennas N_{t}, receive antennas N_{r}, subcarriers N_{c}, and real/imaginary parts) are flattened into a one-dimensional vector. The vectors from the P time steps are then concatenated in chronological order to form the frequency domain sequence $\mathbf{X}_{\mathrm{f}} \in \mathbb{R}^{P\times M}$ and the time-domain sequence $\mathbf{X}_{\mathrm{d}} \in \mathbb{R}^{P\times M}$, where $M = 2\times N_{\mathrm{r}}\times N_{\mathrm{t}}\times N_{\mathrm{c}}$.

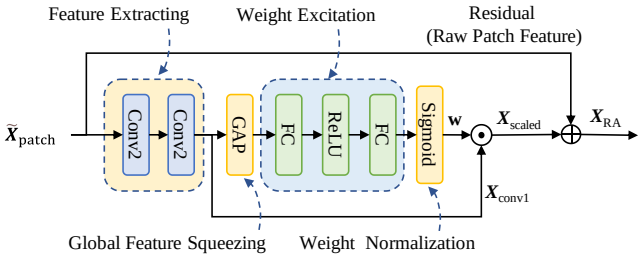


Fig. 3. An illustration of CSI feature module.

To ensure that the CSI features from both the frequency and time domains are weighted equally during LLM training and prevent a single dimension from dominating the model learning due to excessively large values, Z-score normalization is applied to normalize the data from these two domains. The normalized value is computed as:

$$\bar{\mathbf{X}}_{\text{in}} = \frac{\mathbf{X}_{\text{in}} - \mu_{\text{in}}}{\sigma_{\text{in}}} \tag{11}$$

where $\mathbf{X}_{\text{in}}$ denotes the input frequency or time domain CSI sequence, μ_{in} and σ_{in} are its mean and standard deviation, respectively. To match the dimensionality for subsequent processing, the normalized sequences are rearranged into non-overlapping patches, resulting in $P' = P / N \in \mathbb{Z}$ patches, resulting in $\tilde{\mathbf{X}}_{\text{patch}} \in \mathbb{R}^{P' \times N \times M}$, where N is the patching size.

Next, CSI attention weights are added to the time and frequency sequences. As shown in Fig. 3, the main process consists of four steps: feature extraction, global information compression, attention weight generation, and feature weighting enhancement. The input is the reordered CSI sequence $\tilde{\mathbf{X}}_{\text{patch}}$. First, two convolutional layers capture the channel correlations between subcarriers and antenna pairs, yielding $\mathbf{X}_{\text{conv1}} = \text{conv2}(\text{conv2}(\tilde{\mathbf{X}}_{\text{patch}}))$. Then, a global average pooling (GAP) layer compresses $\mathbf{X}_{\text{conv1}}$ along the time step dimension to extract global features within each patch. Subsequently, a fully connected (FC) layer reduces the dimensionality of these global features, followed by a ReLU activation function to introduce nonlinearity; another FC layer restores the dimension, and a Sigmoid activation function generates the attention weight matrix $\mathbf{W}$, where all element falls within the range [0, 1], representing the importance of different subcarriers, antenna pairs, and real/imaginary components. This process is formulated as:

$$\mathbf{W} = \text{Sigmoid}\Big(\text{FC}\Big(\text{ReLU}\Big(\text{FC}\Big(\text{GAP}\big(\mathbf{X}_{\text{conv1}}\big)\Big)\Big)\Big)\Big). \tag{12}$$

Through weight application and residual connection, the attention enhanced output of the CSI is obtained:

$$\mathbf{X}_{\text{RA}} = \mathbf{W} \odot \mathbf{X}_{\text{conv1}} + \tilde{\mathbf{X}}_{\text{patch}} \tag{13}$$

where $\odot$ denotes element-wise multiplication, and $\mathbf{W}$, $\mathbf{X}_{\text{conv1}}$, $\tilde{\mathbf{X}}_{\text{patch}}$ share the same dimension of $\mathbb{R}^{P' \times N \times M}$.

After passing through the CSI attention module, the outputs of $\mathbf{X}_{\text{f}}$ and $\mathbf{X}_{\text{d}}$ are fused and projected to $\mathbf{X'}_{\text{fd}} \in \mathbb{R}^{F \times P'}$, where F is the feature dimension of the pre-trained LLM. To distinguish between patches at different positions, we introduce a fixed positional embedding module based on the sinusoidal encoding principle. It generates a positional embedding tensor $\mathbf{X}_{\text{PA}} \in \mathbb{R}^{F \times P'}$, and the formulation is defined as [46]:

$$\mathbf{X}_{\text{PA}}(i,l) = \begin{cases} \sin\left(l \Big/ 1000^{\frac{i}{F}}\right), & \text{if } i \text{ is even} \\ \cos\left(l \Big/ 1000^{\frac{i}{F}}\right), & \text{if } i \text{ is odd} \end{cases} \tag{14}$$

where $i \in \{1,2,\ldots,F\}$ denotes the feature dimension index, $l \in \{1,2,\ldots,P'\}$ denotes the patch position index. The final output of the preprocessing and embedding module is obtained as $\mathbf{X}_{\text{EB}} = \mathbf{X'}_{\text{fd}} + \mathbf{X}_{\text{PA}} \in \mathbb{R}^{F \times P'}$. Notably, an adaptive weighting mechanism is omitted to preserve the original positional information of $\mathbf{X}_{\text{PA}}$, which is critical for accurately reconstructing CSI in OFDM systems, as positional features directly correspond to the temporal order of channel variations. This design choice mitigates the potential overfitting of these essential features during the network training.

4.2 Pre-trained Lightweight LLM

To balance the accuracy and efficiency of CSI sequence prediction, we selected Qwen2.5-math-1.5B as the base lightweight LLM and enhanced it using KD. The choice of Qwen2.5-math-1.5B is based on two primary reasons: first, its parameter count is moderate, aligning with our resource constraints; second, its causal decoder architecture perfectly matches the dependent nature of CSI sequences, enabling effective utilization of historical information for future forecasting. To further enhance its capability to capture the temporal evolution and multipath propagation characteristics of CSI, we employed KD to transfer generalization knowledge from a larger pre-trained model to this lightweight one.

A. KD for Model Compression

KD is a model compression technique whose core concept involves transferring knowledge from a high performance large model (termed the teacher model) to a smaller model (termed the student model). This transfers the teacher's generalization capabilities to the student, thereby enhancing the student model's performance while achieving model compression. The specific workflow is illustrated in Fig. 4.

First, the teacher model is trained on a large-scale dataset D_{gen} to achieve superior performance. Its Softmax layer is configured with a temperature parameter T_{tem} scaling the

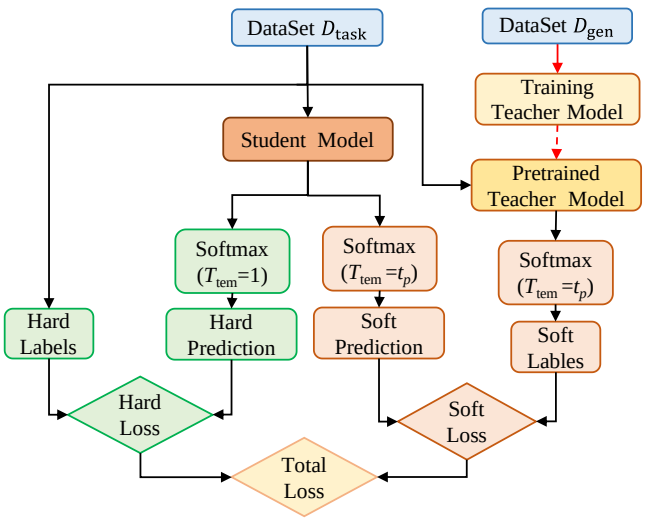


Fig. 4. An illustration of the KD workflow.

Softmax output. Soft labels (probabilistic outputs) are generated to encode the teacher's knowledge about the channel prediction task. Next, the task-specific dataset D_{task} is fed into the student model. The student model produces two parallel predictions: a soft prediction and a hard prediction. The soft prediction (obtained via Softmax with the same temperature T_{tem} as the teacher model) is used to compute the soft loss for minimizing the divergence between the student's and teacher's probabilistic outputs. The hard prediction (obtained via Softmax with temperature $T_{\text{tem}} = 1$) is used to compute the hard loss for matching the student's output to the hard labels in D_{task}, ensuring task-specific fitting. Finally, the total loss is the weighted combination of the soft loss and hard loss. During KD training, the student model's parameters are optimized by minimizing the total loss, which reduces the degree of output discrepancy between the student and teacher, transferring the teacher's generalization knowledge to the student. The result is a distillation-optimized pre-trained model adapted for the channel prediction task.

B. LoRA-Driven Fine-Tuning

To precisely adapt the model to the CSI prediction task, we perform parameter efficient fine-tuning on the distilled model using LoRA. The core logic of this fine-tuning process is illustrated in Fig. 5. Within grouped query attention (GQA) module, LoRA matrices are introduced into the projection layers for the query (**Q**) and value (**V**). Taking the query projection as an example, let the original pre-trained weight be $\mathbf{W}_{\text{q}} \in \mathbb{R}^{F\times F}$, which remains frozen during fine-tuning. LoRA introduces two trainable low-rank matrices, $\mathbf{A}_{\text{q}} \in \mathbb{R}^{F\times r}$ and $\mathbf{B}_{\text{q}} \in \mathbb{R}^{r\times F}$. The Query projection is then updated as follows:

$$\mathbf{Q} = \left(\mathbf{W}_{\text{q}} + \mathbf{A}_{\text{q}}\mathbf{B}_{\text{q}}\right)\mathbf{X} \tag{15}$$

where $\mathbf{X} \in \mathbb{R}^{B\times P'\times F}$ is the input tensor, $r \ll F$ is the rank of the LoRA adaptation, B is the batch size, and P' denotes the sequence length.

Similarly, the value projection is updated as:

$$\mathbf{V} = \left(\mathbf{W}_{\text{v}} + \mathbf{A}_{\text{v}}\mathbf{B}_{\text{v}}\right)\mathbf{X}. \tag{16}$$

The projection weight for the key (**K**), $\mathbf{W}_{\text{k}} \in \mathbb{R}^{F\times F}$, remains frozen and relies solely on the pre-trained knowledge. This design reduces the number of trainable parameters from $O(F^2)$ to $O(2Fr)$, significantly lowering the fine-tuning cost for while preserving the model's capacity to fit channel characteristics.

The output of the LoRA fine-tuned GQA module is then fused via addition and embedded with positional information using the rotary position embedding (RoPE) technique. RoPE encodes positional information by applying 2×2 rotation matrices to pairwise feature dimension groups. The rotation matrix for the i-th group is defined as [46]:

$$R_i(m) = \begin{bmatrix} \cos(m\theta_i) & -\sin(m\theta_i) \\ \sin(m\theta_i) & \cos(m\theta_i) \end{bmatrix} \tag{17}$$

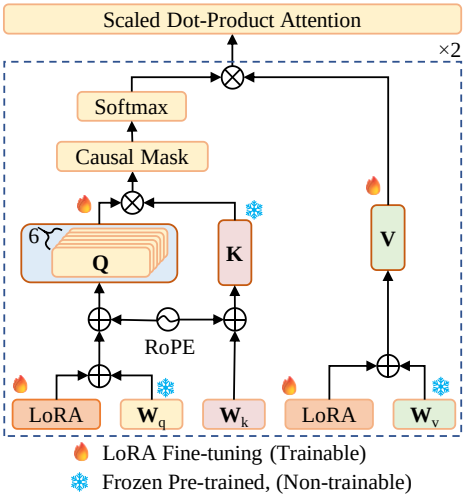


Fig. 5. LoRA fine-tuning process for the GQA module.

where $m \in \{1,2,\ldots,P'\}$ denotes the position index, and $m\theta_i$ represents the rotation angle corresponding to position m. The rotational base frequency θ_i is expressed as [46]:

$$\theta_i = 10000^{-\frac{2(i-1)}{M}}, \quad i = 1,2,\ldots,\frac{M}{2} \tag{18}$$

where $M = 2 \times N_{\text{r}} \times N_{\text{t}} \times N_{\text{c}}$.

C. Causal Mask

To enforce the auto-regressive temporal constraint in CSI prediction, a causal mask is applied after the dot-product operation between **Q** and **K**. Let the attention score matrix be $\mathbf{S} = \mathbf{Q}\mathbf{K}^{\text{T}}/\sqrt{d_{\text{k}}}$, where d_{k} is the key dimension. The causal mask $\mathbf{M} \in \{0, -\infty\}^{P'\times P'}$ is defined as [47]:

$$M_{m,n} = \begin{cases} 0, & m \geq 0 \\ -\infty, & m < n \end{cases} \tag{19}$$

where $m,n \in \{1,2,\ldots,P'\}$, m denotes the current position, and n denotes the attended position. The masked attention score is then computed as **S'** = **S** + **M**. Finally, the attention weights **A** are obtained by applying the Softmax function: **A** = Softmax(**S'**). This operation ensures that when predicting the CSI at the current time step, the model can only attend to information from historical and current time steps, thereby preventing any information leakage from future states.

4.3 Output Processing

The output layer converts the features from a text-like representation to the CSI feature space. The process consists of three steps. First, a two-layer linear mapping is applied to transform the dimension:

$$\mathbf{X}_{\text{LN}} = \text{FC}\left(\text{FC}\left(\mathbf{X}^{\text{LLM}}\right)\right) \in \mathbb{R}^{2N_{\text{t}}N_{\text{t}}N_{\text{c}}\times L} \tag{20}$$

where FC(·) denotes a fully connected linear layer, and L is the prediction horizon length. Subsequently, $\mathbf{X}_{\text{LN}}$ is reshaped

into a tensor $\mathbf{X}_{\mathrm{Re}} \in \mathbb{R}^{2\times N_t\times N_t\times N_c\times L}$. The two feature maps along the first dimension correspond to the real and imaginary components, respectively. Finally, denormalization is performed using the previously calculated statistics as $\mathbf{X}_{\mathrm{De}} = \sigma_{\mathrm{in}}\mathbf{X}_{\mathrm{Re}} + \mu_{\mathrm{in}}$. The output CSI is then reconstructed in complex form by combining the real and imaginary components:

$$\hat{\mathbf{H}} = \mathbf{X}_{\mathrm{De}}[1,:,:,:,:] + \mathrm{j}\mathbf{X}_{\mathrm{De}}[2,:,:,:,:]. \tag{21}$$

where $\mathrm{j} = \sqrt{-1}$ represents the imaginary unit. This yields the final predicted CSI sequence in frequency domain.

5. Experimental Results and Analysis

This section presents experimental results to validate the performance of our proposed lightweight LLM-based channel prediction method. Experiments are implemented on a server equipped with one NVIDIA RTX 4090 D (24 GB GPU memory), Intel Xeon Platinum CPU and 754 GB physical RAM under Ubuntu 22.04 OS. For software configuration, we adopt Python 3.12.3 and PyTorch 2.3.0 with CUDA 12.1. The LoRA-based fine-tuning is realized via parameter-efficient fine-tuning (PEFT) 0.19.1. Auxiliary packages including Transformers, h5py, hdf5storage, einops, NumPy and Matplotlib are utilized for dataset processing and result visualization.

5.1 Experimental Setup

We first configure typical TDD/FDD channel environments with varying UE speeds based on the channel model to generate original CSI datasets. The constructed dataset is split into training, validation and test subsets for full-shot and few-shot experimental settings. Our LLM-based model is optimized via LoRA fine-tuning with most backbone parameters frozen, and prediction accuracy is quantified by NMSE for subsequent performance comparison against traditional methods.

The dataset in this paper is generated based on the 3GPP standardized channel model using the stochastic channel modeling and simulation tool QuaDRiGa[1] to produce time-varying CSI data. Since the experiments in this section focus on validating the prediction accuracy for multi-antenna channels, the BS is configured with a UPA providing rich spatial-dimensional information, including azimuth and elevation angles, to generate complex channel characteristics with multiple paths and scenarios. This ensures sufficient training and testing data for the CSI prediction model.

To concentrate the experimental analysis on the prediction accuracy of multi-antenna CSI at the base station side, the system is configured as a multiple-input single-output OFDM architecture. The BS is equipped with a 4×4 dual-polarized UPA (4 elements per dimension), while the UE employs a 1×4 ULA ($N_r = 4$). The antenna element spacing is set to half-wavelength (a standard configuration for MIMO arrays) to enhance array gain.

The simulation scenarios include 3GPP Urban Macrocell (UMa) and non-line-of-sight (NLOS) environments, modeling a complex channel with 21 scattering clusters, each containing 20 multipath components. The UE movement follows a linear trajectory model with initial coordinates randomly distributed within the coverage area. The dataset is partitioned into 8,000 training segments and 1,000 validation segments, with terminal speeds uniformly sampled between 10 km/h and 100 km/h. The test set comprises 10 equally spaced speed test points (at 10 km/h intervals), each containing 1,000 segments.

In terms of the operating frequency band, both uplink and downlink channels are configured with a bandwidth of 8.64 MHz, divided into 48 resource blocks (each corresponds to 180 kHz subcarriers). For the duplexing scheme, the center frequency of the uplink is set to 2.4 GHz for both TDD and FDD modes, where FDD achieves uplink-downlink isolation through adjacent frequency bands. The channel is assumed to be quasi-static over the $P + L = 20$ consecutive frames considered for prediction ($P = 16$ historical steps and $L = 4$ future steps), with a time interval of 0.5 ms. The patch size and CSI attention processing size are both set to 4. And the kernel size of all convolutional layers is set to 3×3. The DeepSeek-R1-Distill-Qwen-1.5B model (a lightweight pre-trained language model) is used as the base LLM, with a feature dimension of 768.

A comparative framework consisting of traditional prediction algorithms and deep neural network architectures is constructed for baseline selection, which specifically includes:

- No-Prediction Scheme: This strategy employs a zero-prediction approach by directly mapping the latest uplink CSI. While it avoids prediction errors, it fully retains the information delay caused by channel aging, providing a crucial baseline reference for evaluating the effects of channel time-variation.
- CNN [27] Predictor: Innovatively reconstructs the time-frequency domain CSI matrix into a two-dimensional feature map and constructs a deep architecture comprising ten 3×3 convolutional layers. This design captures the joint correlations along the time-frequency dimensions through a local perception mechanism, while the deep convolutional stacking enables hierarchical feature extraction.
- GRU [24]: As a lightweight variant of LSTM, it employs synergistic operations of update and reset gates to reduce the number of model parameters while maintaining comparable prediction performance. In the channel prediction task, the GRU is configured with four layers.

[1] https://quadriga-channel-model.de/

- LSTM [25]: Built upon the standard RNN, it introduces memory cells and a triple-gating mechanism (input gate, forget gate, output gate). The 4-layer stacked structure significantly enhances the ability to model long-term dependencies in time-varying channel characteristics.
- RNN [22]: A 4-layer recurrent architecture is constructed using the classical Elman network structure, which transmits temporal information through hidden states. However, its practical prediction depth is limited due to the vanishing gradient problem.

PAD [19] Predictor: As an advanced representative of model-driven methods, this algorithm captures the dynamic characteristics of channel variations by constructing an 8-order autoregressive model. Its innovation lies in employing a sliding window mechanism to process 15 historical CSI samples, with prediction parameters dynamically updated using the least squares method. This design is particularly suitable for channel tracking in high-speed mobility scenarios within TDD systems, as the high-order modeling effectively overcomes the tracking lag problem caused by rapid time-varying channels.

To ensure fairness, all comparative experiments are conducted with identical hardware and environmental conditions. Furthermore, the hyperparameters for each method are optimized using the same tuning strategy to compare them at their respective best performance levels. The key simulation parameters are summarized in Tab. 2.

To comprehensively evaluate the performance of the proposed scheme, we employ the NMSE as the metric for assessing model performance. NMSE is a widely used performance indicator, commonly utilized to measure the error between estimated values and true values. Through normalization processing, it eliminates the influence of dimensions, facilitating result comparison across different scenarios and directly reflecting the accuracy of channel prediction. Therefore, in the experiments of this paper, the mean square error is adopted as the primary indicator.

$$\text{Loss} = \frac{\left\|\hat{\mathbf{H}} - \mathbf{H}\right\|_{\text{F}}^2}{\left\|\mathbf{H}\right\|_{\text{F}}^2}. \tag{24}$$

Paremeter	Value
Channel model	QuaDRiGa (3GPP TR 38.901 GSCM, UMa NLOS)
Center frequency	2.4 GHz
Bandwidth	8.64 MHz (48 RB×180 kHz)
Duplexing	TDD and FDD
BS antenna	4×4 dual-polarized UPA
UE antenna	1×4 ULA
Speed range	10–100 km/h
Training samples	8,000
Validation samples	1,000
Test samples	10×1,000
Historical time steps	16
Future time steps	4
Time interval	0.5 ms
Base LLM	DeepSeek-R1-Distill-Qwen-1.5B
Hidden dimension	1,536
LLM layers	6 (first 6 of 28)
Feature extraction	Time-frequency dual-domain, 4 Res_block each
Batch size	256
Epoch	200
Optimizer	Adam
Learning rate	0.0001
LoRA configuration	$r = 8$, $\alpha = 16$
KD temperature T / weight α	4.0 / 0.5

Tab. 2. Key simulation parameter settings.

5.2 Performance Evaluation

A. Performance under Time-Varying Speed

The impact of UE mobility speed on channel prediction accuracy in TDD systems is compared in Fig. 6, with a specific focus on our LLM-based CSI prediction model (labeled as DS-1.5B). Experimental results reveal distinct performance patterns across different methods as UE speed increases from 10 km/h to 100 km/h.

The NMSE of traditional baseline methods, e.g., CNN, GRU, LSTM and RNN, generally approaches 0.1 at high speeds (≥ 80 km/h). For instance, at 60 km/h, CNN and GRU yield NMSE values of 4.4×10^{-2} and 7.1×10^{-2}, respectively. This degradation stems from Doppler spread, which shortens the channel coherence time under high mobility conditions, making it challenging for these methods to capture rapidly time-varying channel dynamics. No-prediction method (labeled as np) directly uses the latest CSI as prediction, exhibits significant errors (NMSE > 0.1 even at 20 km/h) and shows a steep increase in error with speed. It fails to account for channel temporal correlations, making it unsuitable for dynamic scenarios.

The proposed DS-1.5B model demonstrates consistent superiority over all baselines across the entire speed range. This performance gain stems from the time-frequency dual-domain feature decoupling mechanism, which explicitly models subcarrier-frequency correlations and antenna-spatial dependencies, enabling accurate capture of CSI variations under mobility. Even at extreme mobility (100 km/h), DS-1.5B maintains an NMSE below 5.7×10^{-2}, achieving 29% and 43% reduction over CNN and GRU, respectively. Notably, compared to the state-of-the-art PAD method, our model achieves a 76% NMSE reduction at 80 km/h, which can be attributed to the strong generalization capability enabled by parameter-efficient fine-tuning via LoRA and validated its robust modeling capability for rapidly time-varying channels.

In FDD systems, downlink CSI must be fed back to the base station via the uplink, a process that introduces inherent feedback delays (typically ranging from several milliseconds to tens of milliseconds). Since channel aging in FDD is primarily induced by this feedback delay rather than variations in UE mobility speeds, the performance fluctuation of the NMSE across different speed scenarios is significantly mitigated, as examined in Fig. 7.

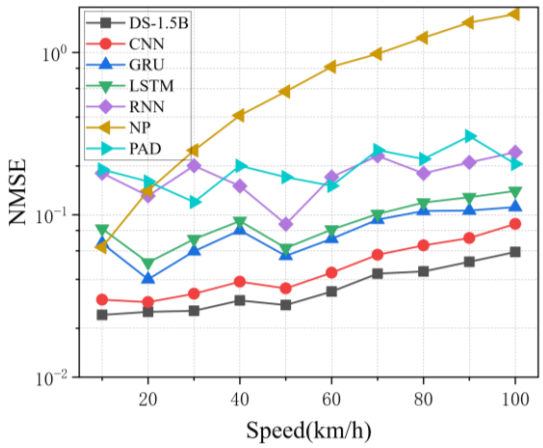


Fig. 6. NMSE of each model under different UE speeds in the TDD system.

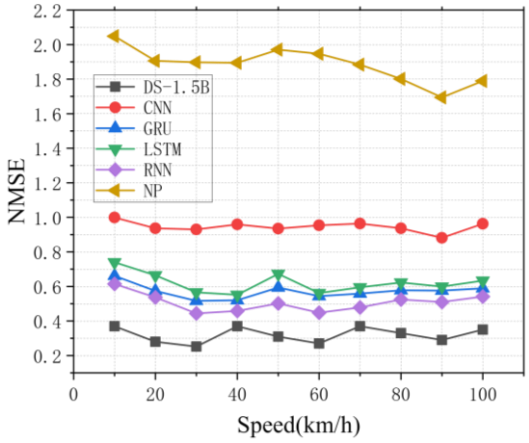


Fig. 7. NMSE of each model under different UE speeds in the FDD system.

DS-1.5B exhibits remarkable stability across the entire speed range, with NMSE consistently below 0.4, validating the robustness of the proposed end-to-end lightweight LLM-driven framework under dominant feedback delay induced aging. This outstanding stability stems from the time-frequency dual-domain feature decoupling mechanism, which explicitly model's frequency-domain subcarrier correlations to compensate for CSI distortion accumulated during feedback delay, rather than relying solely on temporal evolution. By decoupling and reconstructing channel dynamics in the frequency domain, DS-1.5B effectively mitigates aging effects even when temporal information is compromised by feedback latency.

Traditional baselines including CNN, GRU, LSTM, and RNN show significantly higher NMSE (~0.6, 50% higher than DS-1.5B), suffering from limited capability in exploiting frequency-domain correlations to counteract feedback-delay-induced errors. The no-prediction method fails completely (NMSE ~2) due to its inability to account for channel changes during feedback.

To further evaluate the transmission performance, we compare the equal-gain-combining-based (EGC-based) downlink spectral efficiency (SE) of all considered methods under TDD and FDD systems, with detailed numerical results listed in Tabs. 3 and 4, respectively. At the transmitter side, the base station adopts maximum ratio transmission (MRT) precoding with uniform power allocation across all subcarriers. The downlink signal-to-noise ratio (SNR) is fixed to 10 dB, defined as the ratio of transmit power to noise power. The achievable SE is calculated by summing the Shannon capacity over 48 subcarriers. With perfect channel state information (CSI), the theoretical EGC SE upper bound reaches 7.91 bit/s/Hz.

From Tab. 3 and Tab. 4, we know that all methods yield SE values lower than the perfect-CSI EGC upper bound of 7.91 bit/s/Hz, and the cross-band channel decoupling of FDD introduces extra prediction difficulty, leading to universally degraded SE relative to TDD. In TDD mode, DS-1.5B attains the highest SE across all moving speeds, delivering an average of 7.464 bit/s/Hz (94.3% of the ideal bound) and a 24.2% SE gain over the no-prediction baseline at 90 km/h. For FDD scenarios, DS-1.5B still maintains clear superiority with an average SE of 7.067 bit/s/Hz, outperforming the second-best RNN by 13% and the no-prediction scheme by 21.4%, while CNN suffers severe performance deterioration due to its limited capacity to capture cross-band channel variations. These observations verify that accurate spatio-temporal-frequency channel prediction is critical for sustaining high spectral efficiency, especially under high mobility and FDD cross-frequency constraints.

B. Few-Shot Performance

To reduce the resource overhead of CSI acquisition and model training, developing few-shot prediction methods is of great significance for the rapid model deployment. In this paper, an experimental scenario with low data volume is constructed by using only 10% of the dataset for network training, and the few-shot learning performance of different models is systematically evaluated.

Speed(km/h)	PAD	CNN	GRU	LSTM	RNN	NP	DS-1.5B
30	7.659	7.678	7.599	7.558	7.393	7.344	7.692
60	7.487	7.526	7.433	7.413	7.270	6.513	7.548
90	7.098	7.085	6.767	6.802	6.682	5.759	7.152
Average	**7.415**	**7.429**	**7.266**	**7.258**	**7.115**	**6.539**	**7.464**

Tab. 3. The SE performance comparison for TDD systems (Maximum achievable SE: 7.91 bit/s/Hz).

Speed(km/h)	RNN	GRU	LSTM	NP	CNN	DS-1.5B
30	6.402	6.216	6.215	5.868	4.798	7.191
60	6.334	6.159	6.198	5.645	4.566	7.022
90	5.995	5.740	5.798	5.946	4.507	6.989
Average	**6.244**	**6.038**	**6.071**	**5.820**	**4.623**	**7.067**

Tab. 4. The SE performance comparison for FDD systems (Maximum achievable SE: 7.91 bit/s/Hz).

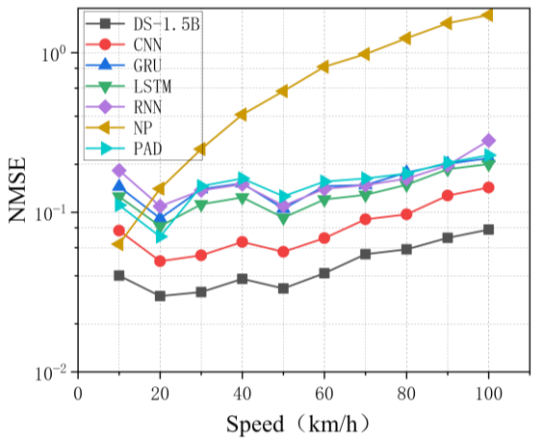


Fig. 8. NMSE of each model under different UE speeds in the TDD system (few-shot).

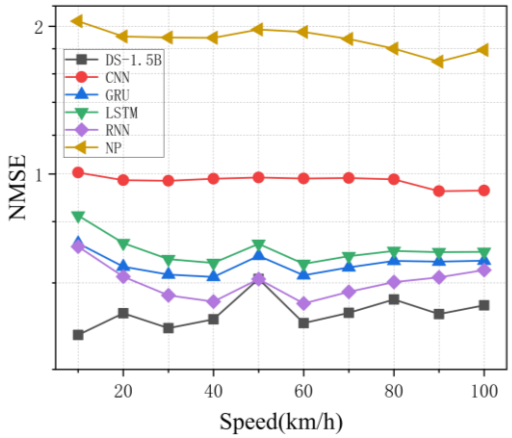


Fig. 9. NMSE of each model under different UE speeds in FDD system (few-shot).

As shown in Fig. 8, with the UE speed increases from 10 km/h to 100 km/h, the NMSE of the DS-1.5B increases from 4×10^{-2} to 7.8×10^{-2}, yet it still maintains a significant and consistent advantage over other models. Compared with full-sample training, such stable and superior prediction capability is attributed to the model's tailored time-frequency dual-domain feature extraction module. This module can efficiently mine robust multi-dimensional channel features even with limited training samples, enabling the model to well adapt to the fast time-varying channel conditions caused by high user mobility.

Traditional baseline methods including CNN, GRU, LSTM and RNN present relatively high NMSE values and less stability. For instance, the NMSE values of GRU and LSTM are notably higher than that of DS-1.5B under all speed conditions. Their inferior performance in the few-shot scenario stems from excessive dependence on large-scale labeled datasets to learn effective channel patterns, making them prone to severe overfitting or underfitting when training data is scarce. In contrast, the proposed framework integrates LoRA with KD from pre-trained large language models, which drastically cuts down computational complexity and the number of trainable parameters, realizing efficient few-shot learning and reliable generalization performance even with sparse training data.

The no-prediction method achieves relatively low NMSE at low speeds, but its performance deteriorates sharply as the UE speed increases. This method fails to capture the inherent temporal evolution correlations of wireless channels, especially in the few-shot scenario with limited historical CSI data, resulting in drastic prediction errors under high mobility conditions. Although the PAD method outperforms traditional deep learning baselines, it still lags behind DS-1.5B obviously. The performance degradation of PAD in the few-shot scenario reveals its limited generalization ability from sparse data samples. By contrast, the knowledge distillation architecture of DS-1.5B enables effective utilization of pre-trained prior knowledge even in few-shot cases, which is a core advantage of the LoRA-enhanced few-shot learning strategy; this design maintains strong generalization while lowering resource overhead, making the framework suitable for resource-constrained compact devices.

Furthermore, the distinct separation between DS-1.5B's performance curve and those of other baselines in Fig. 8 fully validates its strong generalization capability in TDD systems with limited training samples. Even as the UE speed continues to rise, the model can still sustain superior CSI prediction accuracy, which demonstrates its strong robustness to both high-mobility-induced channel variations and data scarcity constraints.

In Fig. 9, we further verified the few-shot learning performance of the proposed method in FDD mode. Traditional baseline methods including CNN, GRU, LSTM and RNN exhibit conspicuous performance degradation under this scenario. Specifically, CNN achieves an extremely high NMSE value close to 1, which indicates that it is completely incapable of completing accurate CSI prediction tasks in FDD systems with limited training samples. GRU, LSTM, and RNN achieve NMSE values ranging from 6.5×10^{-1} to 7.5×10^{-1}, which are still far higher than that of DS-1.5B. Their inferior performance is mainly attributed to heavy dependence on massive labeled datasets to capture effective channel variation patterns, rendering them prone to serious underfitting when training data is scarce. No-prediction method performs the worst, with an NMSE value of around 2, which is far beyond the acceptable range for practical communication. In FDD systems, the inherent frequency-band separation between uplink and downlink channels eliminates channel reciprocity, and coupled with limited training samples, this naive non-prediction strategy becomes completely ineffective, as it fails to track and compensate for channel state changes during the CSI feedback process.

Model	NMSE (TDD 100 km/h)	Trainable Params
PAD	0.206	0
RNN	0.243	0.31 M
GRU	0.112	0.86 M
LSTM	0.140	1.14 M
CNN	0.088	3.15 M
DS-1.5B	0.059	19.66 M

Tab. 5. Performance and parameter comparison.

Although the non-reciprocity of FDD channels induced by frequency division duplexing substantially increases the difficulty of prediction, DS-1.5B still maintains a significant advantage within the UE speed range of 10–100 km/h. It demonstrates remarkable stability, with NMSE consistently maintaining at approximately 6×10^{-1} across the entire speed range. This stable performance stems from two core designs of the proposed framework: the time-frequency dual-domain feature decoupling mechanism that enables robust extraction of multi-dimensional channel features, and the integration of LoRA with pre-trained LLM knowledge distillation that realizes efficient few-shot learning with limited training data. In FDD systems, feedback delay is identified as the dominant factor affecting CSI prediction error, and DS-1.5B's capability to explicitly model the temporal correlations introduced by such delay ensures reliable and accurate prediction even in data-scarce scenarios.

Notably, the minimal NMSE fluctuation of DS-1.5B as UE speed increases from 10 km/h to 100 km/h further validates the conclusion that feedback delay, rather than UE mobility, is the dominant factor affecting prediction error in FDD systems. Even under the few-shot condition with sparse training data, the knowledge-distilled architecture of DS-1.5B enables strong generalization ability, maintaining stable and superior prediction performance throughout the tested speed range.

Comprehensive experimental results demonstrate that the proposed LLM-based CSI prediction algorithm can effectively reduce the model's dependence on large-scale labeled datasets, while retaining stable and high-precision channel prediction performance. This unique advantage makes the framework highly suitable for rapid deployment in practical communication systems with constrained computing resources, and provides solid feasibility verification for engineering applications of high-mobility wireless communication.

To further demonstrate the practical advantages of DS-1.5B over traditional methods, Table 5 compares the prediction accuracy and parameter efficiency across all baselines. Under the TDD full-shot setup with a UE speed of 100 km/h, DS-1.5B obtains an NMSE of 0.059, corresponding to a reduction of 33.0%, 57.9%, 71.4% and 75.7% against CNN (0.088), LSTM (0.140), PAD (0.206) and RNN (0.243), respectively. Under few-shot TDD at 100 km/h, DS-1.5B achieves an NMSE of 0.078, while CNN reaches 0.143 and LSTM degrades to 0.201. In FDD full-shot mode, DS-1.5B maintains NMSE between 0.25 and 0.37 across all speeds, whereas CNN fluctuates between 0.88 and 1.00. In FDD few-shot mode, DS-1.5B maintains NMSE between 0.47 and 0.61, while CNN fluctuates between 0.92 and 1.01. In terms of computational efficiency, DS-1.5B requires only 0.15M LoRA adapter parameters (19.66M total trainable parameters), compared to 3.15M for CNN, 1.14M for LSTM, 0.86M for GRU, and 0.31M for RNN. These results confirm that the proposed framework delivers state-of-the-art prediction accuracy while achieving a favorable accuracy-efficiency trade-off suitable for resource-constrained edge deployment.

C. Performance of Zero-Shot Generalization

Model generalization capability stands as a critical indicator for practical real-world deployment of channel prediction models. A well-trained and high-quality model should maintain stable and effective prediction performance in unseen communication scenarios, with no need for extra fine-tuning or retraining data. To verify the transfer adaptability of the proposed framework, this paper further evaluates the zero-shot generalization performance under the TDD system via a zero-shot transfer strategy. Specifically, all models are fully trained on the 3GPP UMa scenario, and then directly transferred to the 3GPP urban microcell (UMi) scenario for testing, with all hyperparameters and network configurations kept unchanged, completely omitting the retraining process to simulate a real zero-shot deployment scenario.

Figure 10 depicts the zero-shot generalization performance of all comparison models transferred from UMa to UMi scenarios in TDD systems, with respect to varying UE speeds. Two key conclusions can be drawn from the experimental results. Firstly, owing to the faster time-varying characteristics of UMi channel environments, the NMSE values of all models show an upward trend as the UE speed increases, but the degree of performance degradation varies drastically among different models. DS-1.5B achieves the minimum NMSE of only 3.3×10^{-2} at 20 km/h and the maximum NMSE of just 1.21×10^{-1} at 100 km/h, maintaining the lowest prediction error across the entire tested speed range. In contrast, traditional baseline models such as CNN and LSTM suffer from severe performance fluctuations and rapid NMSE growth, with CNN's prediction error spiking sharply under high-speed conditions, leading to drastic performance collapse. Secondly, DS-1.5B maintains a stable performance advantage over all baseline models at every speed point. Its low and steady NMSE confirms that the proposed model captures inherent and generalizable channel variation patterns from pre-training, rather than overfitting to scenario-specific features of the UMa environment. Furthermore, the strong robustness of the LLM-driven DS-1.5B model to high-mobility scenarios highlights its practical suitability for real-world TDD systems, where UE speeds

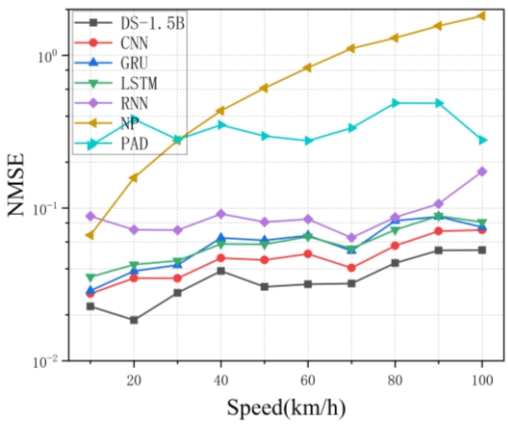


Fig. 10. The generalization ability of the TDD system in the Umi scenario.

often fluctuate widely. In summary, the zero-shot transfer experimental results fully demonstrate that the proposed DS-1.5B model significantly outperforms all benchmark methods in terms of prediction error and stability, validating its superior cross-scenario adaptation capability under varying channel distribution conditions.

6. Conclusions

This paper presents an end-to-end lightweight LLM based channel prediction framework for MIMO-OFDM systems, which fine-tunes the DeepSeek-R1-Distill-Qwen-1.5B architecture, a distilled version of Qwen2.5-Math-1.5B with strong mathematical and logical reasoning capabilities. Benefiting from its compact parameter scale, the model is well suited for deployment on resource-constrained mobile and edge computing devices. By exploiting historical CSI sequences, it can accurately predict future downlink CSI sequences for both FDD and TDD systems, thereby alleviating CSI aging and performance degradation caused by high-mobility propagation conditions. The framework integrates a time-frequency dual-domain input module, followed by preprocessing and position embedding to flexibly adapt the LLM to the CSI prediction task. To enable efficient adaptation to high-mobility MIMO-OFDM scenarios, most model parameters are frozen, and only the attention mechanism is fine-tuned via LoRA. Experimental results confirm DS-1.5B yields NMSE of 0.059 and 0.078 under full-shot and few-shot TDD at 100 km/h, reducing prediction error by 33.0%, 57.9%, 71.4% versus CNN, LSTM and PAD in full-shot TDD and by 45.5%, 61.2% versus CNN, LSTM in few-shot TDD, respectively. For full-range speeds under few-shot FDD, its NMSE is confined within [0.47, 0.61] against severely deteriorated baselines. Besides, the peak NMSE stays as low as 0.053 during zero-shot UMa-to-UMi migration without retraining, demonstrating prominent prediction superiority and robust cross-scenario generalization. With merely 0.15M LoRA adapter parameters among 19.66M total trainable parameters, the framework is readily deployable on resource-constrained edge devices. Future work will focus on validation with real-world channel measurements and further latency optimization. To promote reproducibility, the the source code for this simulation-based study is publicly available at https://github.com/cjx12363/DS-1.5B.

Several directions remain for future investigation to further improve the proposed method. First, the current validation is conducted solely based on QuaDriGa simulation data, and real-world measurement datasets need to be introduced for more practical verification. Second, although the model exhibits promising cross-scenario generalization, its robustness and adaptability under extreme channel conditions (e.g., ultra-high mobility) still require further validation. Third, the overall system design can be further optimized to meet the strict requirements of ultra-low-latency communication applications. These limitations provide clear guidance for our future research and improvement.

Acknowledgments

This work was funded by In-service Doctoral Research Funding Project of Nanjing Institute of Technology, grant number ZKJ202002.

References

[1] PARK, J., SOHRABI, F., GHOSH, A., et al. End-to-end deep learning for TDD MIMO systems in the 6G upper midbands. *IEEE Transactions on Wireless Communications*, 2025, vol. 24, no. 3, p. 2110–2125. DOI: 10.1109/TWC.2024.3516633

[2] BJÖRNSON, E., HOYDIS, J., KOUNTOURIS, M. Massive MIMO systems with non-ideal hardware: Energy efficiency, estimation, and capacity limits. *IEEE Transactions on Information Theory*, 2014, vol. 60, no. 11, p. 7112–7139. DOI: 10.1109/TIT.2014.2354403

[3] LARSSON, E., EDFORS, O., TUFVESSON, F., et al. Massive MIMO for next generation wireless systems. *IEEE Communications Magazine*, 2014, vol. 52, no. 2, p. 186–195. DOI: 10.1109/MCOM.2014.6736761

[4] TRUONG, K. T., HEATH, R. W. Effects of channel aging in massive MIMO systems. *Journal of Communications and Networks*, 2013, vol. 15, no. 4, p. 338–351. DOI: 10.1109/JCN.2013.000065

[5] SHI, J., LI, Z., HU, J., et al. OTFS enabled LEO satellite communications: A promising solution to severe Doppler effects. *IEEE Network*, 2024, vol. 38, no. 1, p. 203–209. DOI: 10.1109/MNET.129.2200458

[6] LIM, B., YUN, W. J., KIM, J., et al. Joint pilot design and channel estimation using deep residual learning for multi-cell massive MIMO under hardware impairments. *IEEE Transactions on Vehicular Technology*, 2022, vol. 71, no. 7, p. 7599–7612. DOI: 10.1109/TVT.2022.3170556

[7] JEE, J., PARK, H. Deep learning-based joint optimization of closed-loop FDD mmwave massive MIMO: Pilot adaptation, CSI feedback, and beamforming. *IEEE Transactions on Vehicular Technology*, 2024, vol. 73, no. 3, p. 4019–4034. DOI: 10.1109/TVT.2023.3327276

[8] ZHENG, J., ZHANG, J., BJÖRNSON, E., et al. Impact of channel aging on cell-free massive MIMO over spatially correlated channels. *IEEE Transactions on Wireless Communications*, 2021, vol. 20, no. 10, p. 6451–6466. DOI: 10.1109/TWC.2021.3074421

[9] BENZINE, W., BEMANI, A., KSAIRI, N., et al. Models, methods, and waveforms for estimation and prediction of sparse time-varying channels. *IEEE Transactions on Wireless Communications*, 2026, vol. 25, p. 9623–9638. DOI: 10.1109/TWC.2025.3644666

[10] PAPAZAFEIROPOULOS, A. K. Impact of general channel aging conditions on the downlink performance of massive MIMO. *IEEE Transactions on Vehicular Technology*, 2017, vol. 66, no. 2, p. 1428–1442. DOI: 10.1109/TVT.2016.2570742

[11] ZHOU, B., YANG, X., MA, S., et al. Low-overhead channel estimation via 3D extrapolation for TDD mmwave massive MIMO systems under high-mobility scenarios. *IEEE Transactions on Wireless Communications*, 2025, vol. 24, no. 4, p. 2797–2813. DOI: 10.1109/TWC.2024.3524911

[12] GE, L., WANG, Z., QIAN, L., et al. Sparsity adaptive compressive sensing based two-stage channel estimation algorithm for massive MIMO-OFDM systems. *Radioengineering*, 2023, vol. 32, no. 2, p. 197–206. DOI: 10.13164/re.2023.0197

[13] GIZZINI, A. K., CHAFII, M. Deep learning based channel estimation in high mobility communications using Bi-RNN networks. In *Proceedings of the ICC 2023 - IEEE International Conference on Communications*. Rome (Italy), 2023, p. 2607–2612. DOI: 10.1109/ICC45041.2023.10278783

[14] PENG, F., ZHANG, S., JIANG, Z., et al. A novel mobility induced channel prediction mechanism for vehicular communications. *IEEE Transactions on Wireless Communications*, 2023, vol. 22, no. 5, p. 3488–3502. DOI: 10.1109/TWC.2022.3219052

[15] SANG, Y., MA, K., WANG, Z., et al. Dual-band super-resolution channel prediction in high-mobility MIMO systems. *IEEE Transactions on Communications*, 2025, vol. 73, no. 6, p. 4409 to 4424. DOI: 10.1109/TCOMM.2024.3511954

[16] KRISTIANI, E., VERMA, V. K., YANG, C. T., et al. Deploying LLM transformer on edge computing devices: A survey of strategies, challenges, and future directions. *AI*, 2026, vol. 7, no. 1, p. 1–37. DOI: 10.3390/ai7010015

[17] CHONG, B., LU, H., NIYATO, D., et al. Large language model-driven channel prediction in cell-free mMIMO systems. *IEEE Journal on Selected Areas in Communications*, 2026, vol. 44, p. 3412–3426. DOI: 10.1109/JSAC.2026.3654883

[18] QWEN AI. Qwen 2.5 Requirements [EB/OL]. [Online] Available at: https://www.qwen-ai.com/requirements/.

[19] YIN, H., WANG, H., LIU, Y., et al. Addressing the curse of mobility in massive MIMO with prony-based angular-delay domain channel predictions. *IEEE Journal on Selected Areas in Communications*, 2020, vol. 38, no. 12, p. 2903–2917. DOI: 10.1109/JSAC.2020.3005473

[20] WANG, X., SHI, Y., XIN, W., et al. Channel prediction with time-varying Doppler spectrum in high-mobility scenarios: A polynomial Fourier transform based approach and field measurements. *IEEE Transactions on Wireless Communications*, 2023, vol. 22, no. 11, p. 7116–7129. DOI: 10.1109/TWC.2023.3247825

[21] MATTU, S. R., THEAGARAJAN, L. N., CHOCKALINGAM, A. Deep channel prediction: A DNN framework for receiver design in time-varying fading channels. *IEEE Transactions on Vehicular Technology*, 2022, vol. 71, no. 6, p. 6439–6453. DOI: 10.1109/TVT.2022.3162887

[22] JIANG, W., SCHOTTEN, H. D. Neural network-based fading channel prediction: A comprehensive overview. *IEEE Access*, 2019, vol. 7, p. 118112–118124. DOI: 10.1109/ACCESS.2019.2937588

[23] LIU, Q., CAO, N., LI, M., et al. Wireless channel state prediction method based on improved adaptive and parameter-free recurrent neural structure. *IEEE Access*, 2022, vol. 10, p. 63329–63338. DOI: 10.1109/ACCESS.2022.3182376

[24] POLAK, L., TURAK, S., SOTNER, R., et al. Exploring deep learning architectures for RF signal classification. In *35th International Conference Radioelektronika*. Hnanice (Czech Republic), 2025, p. 1–6. DOI: 10.1109/radioelektronika65656.2025.11008396

[25] XIAO, Z., ZHANG, Z., CHEN, Z., et al. From data-driven learning to physics-inspired inferring: A novel mobile MIMO channel prediction scheme based on neural ODE. *IEEE Transactions on Wireless Communications*, 2024, vol. 23, no. 7, p. 7186–7199. DOI: 10.1109/TWC.2023.3338419

[26] ZHANG, Y., WU, Y., LIU, A., et al. Deep learning-based channel prediction for LEO satellite massive MIMO communication system. *IEEE Wireless Communications Letters*, 2021, vol. 10, no. 8, p. 1835–1839. DOI: 10.1109/LWC.2021.3083267

[27] SAFARI, M. S., POURAHMADI, V., SODAGARI, S. Deep UL2DL: Data-driven channel knowledge transfer from uplink to downlink. *IEEE Open Journal of Vehicular Technology*, 2020, vol. 1, p. 29–44. DOI: 10.1109/OJVT.2019.2962631

[28] CAO, C., CHEN, M., ZHANG, Y., et al. Deep learning-enhanced channel prediction for XL-MIMO systems. *IEEE Communications Letters*, 2025, vol. 29, no. 6, p. 1260–1264. DOI: 10.1109/LCOMM.2025.3558838

[29] ZHANG, Z., ZHANG, Y., ZHANG, J., et al. Adversarial training-aided time-varying channel prediction for TDD/FDD systems. *China Communications*, 2023, vol. 20, no. 6, p. 100–115. DOI: 10.23919/JCC.fa.2020-0698.202306

[30] ZHOU, T., LIU, X., XIANG, Z., et al. Transformer network based channel prediction for CSI feedback enhancement in AI-native air interface. *IEEE Transactions on Wireless Communications*, 2024, vol. 23, no. 9, p. 11154–11167. DOI: 10.1109/TWC.2024.3379123

[31] JIANG, H., CUI, M., NG, D. W. K., et al. Accurate channel prediction based on transformer: Making mobility negligible. *IEEE Journal on Selected Areas in Communications*, 2022, vol. 40, no. 9, p. 2717–2732. DOI: 10.1109/JSAC.2022.3191334

[32] OU, R., LIU, X., YI, Y. Channel state information prediction using transformer models for high-mobility wireless networks. In *Proceedings of the 2025 International Conference on Electrical Automation and Artificial Intelligence (ICEAAI)*. Guangzhou (China), 2025, p. 925–929. DOI: 10.1109/ICEAAI64185.2025.10956273

[33] KIM, D., GONG, J., KANG, J. MIMO channel prediction via deep learning-based conformal Bayes filter. arXiv Preprint, 2026, p. 1–5. DOI: 10.48550/arXiv.2603.04764

[34] CUI, Y., GUO, J., CAO, Z., et al. Lightweight neural network with knowledge distillation for CSI feedback. *IEEE Transactions on Communications*, 2024, vol. 72, no. 8, p. 4917–4929. DOI: 10.1109/TCOMM.2024.3377724

[35] LIU, B., LIU, X., GAO, S., et al. LLM4CP: Adapting large language models for channel prediction. *Journal of Communications and Information Networks*, 2024, vol. 9, no. 2, p. 113–125. DOI: 10.23919/JCIN.2024.10582829

[36] FAN, S., LIU, Z., GU, X., et al. Csi-LLM: A novel downlink channel prediction method aligned with LLM pre-training. In *2025 IEEE Wireless Communications and Networking Conference*. Milan (Italy), 2025, p. 1–6. DOI: 10.1109/WCNC61545.2025.10978424

[37] LI, Z., YANG, Q., XIONG, Z., et al. Bridging the modality gap: Enhancing channel prediction with semantically aligned LLMs and knowledge distillation. *IEEE Journal on Selected Areas in Communications*, 2026, vol. 44, p. 3382–3396. DOI: 10.1109/JSAC.2025.3647607

[38] HE, J., REN, Z., YAO, J., et al. Sensing-assisted channel prediction in complex wireless environments: An LLM-based approach. *IEEE Wireless Communications Letters*, 2025, vol. 14, no. 12, p. 3857 to 3861. DOI: 10.1109/LWC.2025.3600454

[39] HE, X., LV, Y., CUI, S., et al. Multimodal large language model-aided environment-aware channel prediction and beamforming. *IEEE Network*, 2026, vol. 40, no. 3, p. 229–238. DOI: 10.1109/MNET.2026.3660027

[40] YANG, H., LAMBOTHARAN, S., DERAKHSHANI, M. FAS-LLM: Large language model-based channel prediction for OTFS-enabled satellite-FAS links. *IEEE Journal on Selected Areas in Communications*, 2026, vol. 44, p. 2952–2963. DOI: 10.1109/JSAC.2025.3647013

[41] YU, L., SHI, L., ZHANG, J., et al. ChannelGPT: A large model toward real-world channel foundation model for 6G environment intelligence communication. *IEEE Communications Magazine*, 2025, vol. 63, no. 10, p. 68–74. DOI: 10.1109/MCOM.001.2400780

[42] ZHOU, H., HU, C., YUAN, Y., et al. Large language model (LLM) for telecommunications: A comprehensive survey on principles, key techniques, and opportunities. *IEEE Communications Surveys and Tutorials*, 2025, vol. 27, p. 1955–2005. DOI: 10.1109/COMST.2024.3465447

[43] 3GPP RADIO ACCESS NETWORK WORKING GROUP. *Study on Channel Model for Frequencies from 0.5 to 100 GHz (Release 15)*. 3GPP TR 38.901, Sophia Antipolis, France: 3GPP, 2018.

[44] LIU, L., OESTGES, C., POUTANEN, J., et al. The COST 2100 MIMO channel model. *IEEE Wireless Communications*, 2012, vol. 19, no. 6, p. 92–99. DOI: 10.1109/MWC.2012.6393523

[45] AYACH, O. E., RAJAGOPAL, S., ABU-SURRA, S., et al. Spatially sparse precoding in millimeter wave MIMO systems. *IEEE Transactions on Wireless Communications*, 2014, vol. 13, no. 3, p. 1499–1513. DOI: 10.1109/TWC.2014.011714.130846

[46] SU, J., LU, Y., PAN, S., et al. RoFormer: Enhanced transformer with rotary position embedding. arXiv Preprint, 2021. DOI: 10.48550/arXiv.2104.09864

[47] VASWANI, A., SHAZEER, N., PARMAR, N., et al. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS'17)*. Long Beach (CA, USA), 2017, p. 6000–6010. DOI: 10.5555/3295222.3295349

About the Authors ...

Xiongli RUI was born in Zhejiang Province, China, in 1978. She received the B.E. degree, the M.E. degree from Hohai University, Nanjing, China, in 2000 and 2003, respectively. She received Ph.D. degree from Nanjing University of Post and Telecommunications in 2020. Her major research interests include wireless communication and deep learning.

Rui CHEN (corresponding author) received the B.E. degree, the M.E. degree from Southeast University, Nanjing, China, in 1991 and 1996, respectively. She received Ph.D. degree from Nanjing University of Post and Telecommunications in 2013. Her major research interests include deep learning and wireless communication.

Xiaoyan ZHAO was born in Jiangsu Province, China, in 1986. She received the B.E. degree and the Ph.D. degree from Southeast University, Nanjing, China, in 2008 and 2014, respectively. Her major research interests include signal processing and deep learning.

Jie YANG received her B.E. and M.E. from Lanzhou University of Technology, Lanzhou, China, in 2000 and 2003. She received her Ph.D. from Nanjing University of Posts and Telecommunications in 2015. Her research interests include wireless communication, signal processing and deep learning.

Jinxiang CHEN is a graduate student majoring in Electrical Engineering at Nanjing Institute of Technology. At school, he has been involved in his teachers' research projects. His research interests include electric vehicle charging, space-time prediction and deep learning.