Published by :
https://www.ijert.org/
An International Peer-Reviewed Journal

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 07, NCIRTAI - 2026

Emerging Advances in Artificial Intelligence for Future Wireless Networks

M. Sindhu
Assistant Professor, Dept. of Computer Science and Engg, Sri Raaja Raajan College of Engg & Technology, Karaikudi,India
sindhusrrcetsoftware@gmail.com

Abstract: Intelligent communication is considered one of the mainstream directions for the development of wireless communication post-5G. Its core idea involves integrating artificial intelligence into various layers of wireless communication systems, achieving an organic fusion of wireless communication and AI technologies. Currently, research in this area is rapidly advancing toward the physical layer, though the integration of wireless transmission technologies and AI remains in the early exploratory stages. Focusing on AI-based key wireless transmission technologies, this paper provides a detailed introduction to channel estimation, signal detection, channel state information feedback and reconstruction, channel decoding, and end-to-end wireless communication systems. It elaborates on the latest research progress in this field within the international academic community and further discusses the future development trends of AI-driven wireless transmission technologies.

Keywords: Artificial intelligence; Wireless transmission technology; Deep learning.

1. INTRODUCTION

Since 2010, 5G technology has garnered significant attention from both academia and industry, characterized by its high-dimensional capabilities, high capacity, denser networks, and lower latency. Compared to the commercially deployed 4G systems, 5G wireless transmission speeds have increased by 10–100 times, with peak transmission rates reaching 10 Gbit/s. End-to-end latency has been reduced to the millisecond level, connection density has grown by 10– 100 times, traffic density has improved by 1,000 times, and spectrum efficiency has increased by 5–10 times—all while ensuring a seamless user experience at speeds of up to 500 km/h. Unlike 2G/3G/4G, which were designed primarily for human-to-human communication, 5G was conceived from the outset to interconnect humans and machines as well as machine-to-machine communication. The International Telecommunication Union has outlined eight key 5G performance indicators: peak data rate at base stations, user- experienced data rate, spectrum efficiency, area traffic capacity, mobility, network energy efficiency, connection density, and latency.

So far, 5G has primarily achieved the above indicators through three key dimensions: air interface enhancement, wider spectrum, and network densification. The most representative enabling technologies for these three dimensions are massive MIMO (multiple-input multiple- output), millimeter-wave communication, and ultra-dense networking, respectively. Massive MIMO is regarded as one of the most promising core technologies for 5G due to its numerous advantages, such as improving system capacity, spectral efficiency, user experience data rates, enhancing full- dimensional coverage, and energy savings. However, the development and application of massive MIMO also face many challenges, such as how to effectively acquire channel state information at the base station side for frequency division duplex (FDD) systems that lack uplink-downlink reciprocity. Millimeter waves refer to electromagnetic waves with wavelengths in the millimeter range, corresponding to frequencies of approximately 30–300 GHz. Current wireless communication systems primarily utilize frequency bands ranging from 300 MHz to 3 GHz, with relatively low utilization of the millimeter-wave (mm Wave) spectrum. Millimeter-wave technology enhances network transmission rates by increasing spectrum bandwidth, but it faces significant challenges in practical applications due to propagation path loss, building penetration loss, rain attenuation, and other factors [1].

Additionally, millimeter-wave communication can be effectively integrated with massive MIMO, leveraging the beam forming gains of massive MIMO to compensate for the poor penetration capability of mm Wave signals. Ultra- dense networking (UDN) achieves a more "densified" wireless network deployment by reducing the distance between base stations to tens or even just a dozen meters, significantly increasing site density. This enhances spectrum reuse, network capacity per unit area, and user experience data rates. In summary, massive MIMO exploits ultra-high antenna dimensions to fully utilize spatial resources, millimeter-wave communication leverages ultra-wide bandwidth to boost network throughput, and ultra-dense networking employs ultra-dense base stations to improve spectrum efficiency. Together, these technologies generate massive amounts of wireless big data, providing a valuable data source for future wireless communication systems empowered by artificial intelligence.

In recent years, artificial intelligence—particularly deep learning—has achieved remarkable success in fields such as computer vision, natural language processing, and speech recognition [2]. Researchers in wireless communications are now exploring its application across various layers of communication systems to develop intelligent communication networks. This advancement aims to realize a truly interconnected world of everything (IoE) and meet the ever-growing demand for higher data transmission rates. Therefore, intelligent communication is considered one of the mainstream directions for wireless communication development beyond 5G. Its core concept involves integrating artificial intelligence into various layers of wireless communication systems,

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 07, NCIRTAI - 2026

achieving an organic fusion of wireless communication and AI technologies to significantly enhance system performance. Both academia and industry are actively conducting research in this field. Early-stage research has primarily focused on the application and network layers, mainly applying AI—particularly deep learning—to areas such as wireless resource management and allocation. Currently, research is advancing toward the MAC layer and physical layer, with emerging trends of combining wireless transmission with deep learning at the physical layer. However, these studies remain in the preliminary exploration phase. Although wireless big data presents opportunities for applying artificial intelligence at the physical layer [3], the development of intelligent communication systems remains in an exploratory phase, presenting both opportunities and challenges. Looking back, the evolution of wireless communication systems from 1G to 5G and their tremendous success can be attributed to the establishment and refinement of the wireless transmission theoretical framework based on Shannon's information theory.

A typical wireless communication system consists of a transmitter, wireless channel, and receiver, as illustrated in Figure 1. The transmitter primarily includes modules such as the information source, source coding, channel coding, modulation, and RF transmission. The receiver comprises modules like RF reception, channel estimation and signal detection, demodulation, channel decoding, source decoding, and the information sink. Research on intelligent wireless transmission aims to break away from traditional communication paradigms and achieve significant performance improvements. However, this field faces numerous challenges, and researchers worldwide have only begun preliminary explorations.

This paper primarily reviews the latest research progress in applying deep learning to wireless transmission technologies, including channel estimation, signal detection, channel state information (CSI) feedback and reconstruction, channel decoding, and end-to-end communication systems.

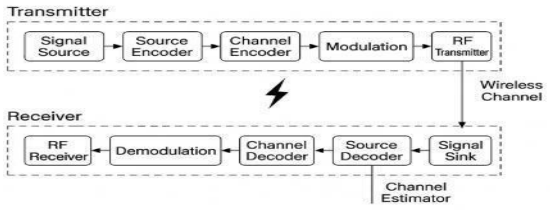

Figure 1. Typical Wirelessss Communication System

Deep Learning Overview

In 1996, Langley defined machine learning as a branch of artificial intelligence aimed at improving system performance by leveraging experiential knowledge. After decades of research since the 20th century, scholars have proposed various algorithms, including logistic regression, decision trees, support vector machines, and artificial neural networks. A pivotal moment came in 2006 when Hinton et al. [4] published a seminal paper in Science, introducing two key in sights: (1) Artificial neural networks with multiple hidden layers exhibit exceptional feature learning capabilities, and (2) the "layer-wise pre-training" approach can effectively overcome training difficulties in deep neural networks. This work marked the advent of modern deep learning research. Subsequently, deep learning has achieved groundbreaking success in speech recognition and computer vision. As an emerging neural network paradigm, deep learning encompasses multiple architectures, including deep neural networks (DNN), convolutional neural networks (CNN), recurrent neural networks (RNN), and generative adversarial networks (GAN). The following sections detail the fundamental structures of these four deep learning architectures.

Deep Neural Networks (DNN)

DNN is also known as a multilayer perceptron. The basic structure of a DNN is shown in Figure 2 and consists of an input layer, multiple hidden layers, and an output layer. Each hidden layer contains multiple neurons, and each neuron is connected to adjacent layers, while neurons within the same layer are not interconnected. A single neuron multiplies each input by a corresponding weight, adds a bias parameter, and then passes the result through a nonlinear activation function. The types of activation functions are listed in Table 1. The back propagation algorithm is an effective method for optimizing DNNs. However, increasing the number of hidden layers and neurons makes the training process more difficult, leading to problems such as vanishing gradients, slow convergence, and convergence to local minima. To address the vanishing gradient problem, new activation functions have been introduced to replace the classical sigmoid function. To improve convergence speed and reduce computational complexity, the classical gradient descent (GD) algorithm has been modified into stochastic gradient descent (SGD), which randomly selects a single sample to compute the loss and gradient at each step. The randomness of SGD can cause significant fluctuations during training. Therefore, mini-batch stochastic gradient descent (mini-batch SGD) is commonly used as a compromise between classical GD and SGD. However, these algorithms may still converge to local optima. To overcome this issue and further improve training speed, several adaptive learning rate algorithms have emerged, such as Adagrad, RMSProp, Momentum, and Adam [4]. If the trained network performs well on the training data but poorly during testing, over fitting has occurred. In such cases, to achieve good performance on both training and testing data, techniques such as regularization and dropout have been proposed.

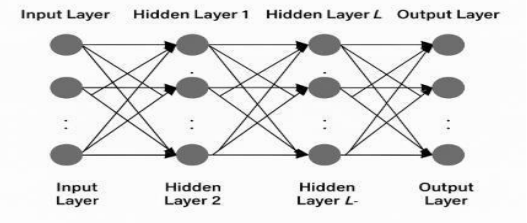

Figure 2. DNN basic structure

Function Name	Activation Function
Sigmoid	$\frac{1}{1+e^{-x}}$
Tanh	$\tanh(x)$
ReLU	$\max(0, x)$

Table 1. Types of Activation Functions

Convolutional Neural Networks (CNN)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 07, NCIRTAI - 2026

The basic structure of a convolutional neural network (CNN) includes an input layer, multiple convolutional layers, multiple pooling layers, fully connected layers, and an output layer, as shown in Figure 3. The convolutional and pooling layers are arranged alternately, meaning each convolutional layer is followed by a pooling layer, and each pooling layer is followed by another convolutional layer, and so on. In a convolutional layer, each neuron in the convolutional kernel is locally connected to its input. It performs a weighted sum of the local input values using corresponding connection weights, adds a bias term, and produces the neuron's output. This process is equivalent to a convolution operation, which is why the network is referred to as a convolutional neural network.

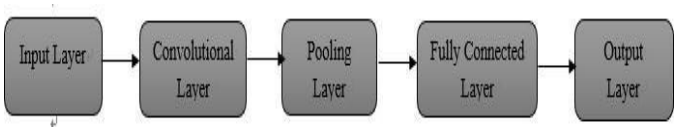

Figure 3. CNN basic structure

Recurrent Neural Networks (RNN)

RNN is a type of neural network designed for modeling sequential data, where the output of the current sequence is also dependent on previous outputs. Specifically, the network remembers information from past moments and applies it in the computation of the current output. In this model, the nodes between hidden layers are no longer disconnected but are connected, and the input to the hidden layers includes not only the input layer but also the output of the hidden layer from the previous time step. Figure 4 is an example of an RNN model. Recurrent neural networks are designed to provide memory for neural networks, as the output depends not only on the current input but also on information available from past or future moments. The time step shown in Figure 4 is 3. Common types of RNNs include Elman networks, Jordan networks, bidirectional RNNs, and Long Short-Term Memory (LSTM).

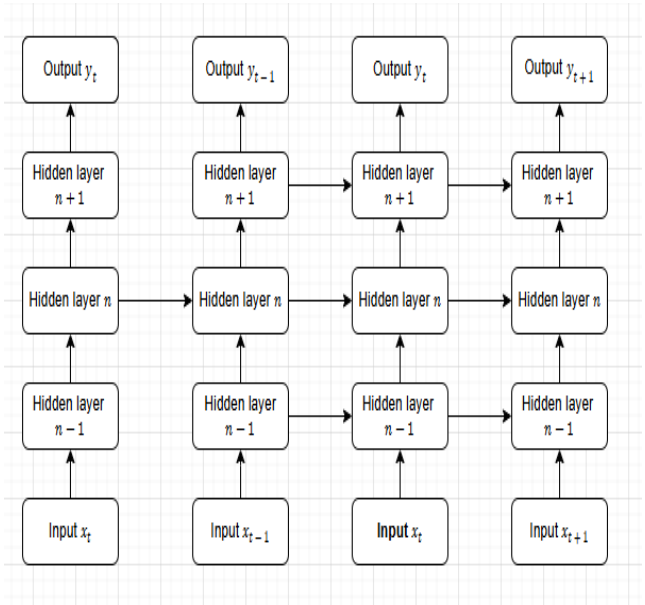

Figure 4. RNN basic structure

Generative Adversarial Network (GAN)

GAN (Generative Adversarial Network) is a novel generative method for distribution learning, aiming to learn a model capable of generating pseudo-samples that resemble data from the real distribution. The structure of GAN is shown in Figure 5 and consists of a generator G and a discriminator D, both implemented using deep neural networks (DNNs). The discriminator is responsible for distinguishing between the pseudo-samples generated by the generator and the real samples from the dataset, while the generator's task is to produce sample data such that the discriminator cannot tell real and fake samples apart. During training, the generator maps input noise z, which follows a prior distribution pz(z), to a sample. Then, samples from the real dataset and those generated by G are collected to train the discriminator D, with the goal of maximizing its ability to distinguish between the two types of data. If the discriminator D successfully classifies real and fake samples, this success is fed back to the generator G, thereby encouraging G to learn to generate samples that are more similar to the real ones. The training process ends when equilibrium is reached, at which point the discriminator D can only make random guesses between real samples and the generated pseudo-samples.

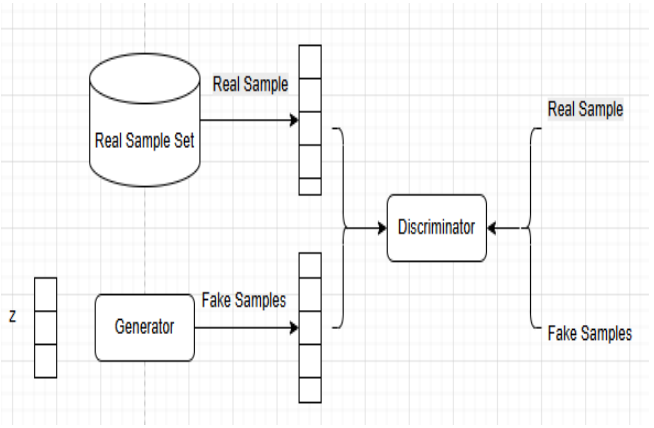

Figure 5. GAN basic structure

Applications of Deep Learning in Wireless Transmission Technology

Channel Estimation

In large-scale MIMO millimeter-wave beamforming scenarios, channel estimation is particularly challenging, especially in environments with dense antenna arrays and limited RF chains at the receiver. Reference [5] proposes the LDAMP network to address this channel estimation problem. This network treats the channel matrix as a two-dimensional image input and integrates a denoising convolutional neural network into an iterative signal reconstruction algorithm for channel estimation. LDAMP is based on the D-AMP algorithm [6] and is composed of L layers of identical structure arranged sequentially. Each layer consists of a denoiser, a divergence estimator, and associated connecting coefficients. The denoiser is implemented using DnCNN, a deep

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 07, NCIRTAI - 2026

convolutional neural network with 20 layers, which plays a decisive role in the LDAMP network. Under unknown noise variance, the DnCNN denoiser can effectively solve the Gaussian denoising problem, offering higher accuracy and faster computation compared to other denoising techniques.

Signal Detection

Reference [8] addresses the problem of signal detection in OFDM (Orthogonal Frequency Division Multiplexing) systems using deep neural networks (DNNs). In traditional OFDM systems, channel estimation and signal detection are treated as two separate functional modules: first, accurate channel state information (CSI) is obtained through estimation, and then the transmitted signal is recovered using the estimated CSI. As shown in Figure 1, the signal recovery process also involves modules such as demodulation. In contrast to conventional wireless communication approaches, Reference [8] considers channel estimation and signal detection as a unified process, employing a DNN to directly learn the mapping from received signals to original transmitted signals. The DNN has an input size of 256, a hidden layer structure of 500–250–120, and an output size of
16. The OFDM system described in the study employs 64 subcarriers and uses quadrature phase shift keying (QPSK) modulation. As a result, the input signal size is 128 bytes, requiring eight identical DNNs to be trained in parallel. After being trained on large volumes of data, the proposed DNN achieves performance comparable to that of the traditional minimum mean square error (MMSE) detection algorithm. In nonlinear OFDM systems such as those without cyclic prefixes or with reduced peak-to-average signal-to-noise ratios the DNN significantly outperforms conventional MMSE methods.

However, this performance gain does not necessarily indicate the soundness of the network design, as the bit error rate (BER) exhibits a saturation effect. This refers to the phenomenon where, as the signal-to-noise ratio (SNR) increases, the BER no longer decreases significantly or ceases to decrease altogether. In practical systems, the issue of nonlinearity remains largely unresolved. It is also worth
noting that training a single DNN requires 20,000 iterations, with each iteration involving 50,000 data samples. Given that eight such networks are needed, the total training time and computational complexity are considerable.

Reference [9] investigates the signal reconstruction problem in MIMO systems and proposes a signal detection algorithm called DetNet. DetNet integrates a gradient descent algorithm into the maximum likelihood framework, thereby forming a deep learning network. To evaluate the robustness of DetNet, two scenarios with known CSI were considered: a time-invariant channel and a time-varying channel with known random variables. Simulation results demonstrate that DetNet outperforms the traditional Approximate Message Passing (AMP) algorithm and achieves performance comparable to the Semi definite Relaxation (SDR) algorithm. Moreover, DetNet offers very high accuracy with significantly reduced computational time, running approximately 30 times faster than conventional algorithms.

Reference [10] addresses the same problem as Reference [9], with both solutions relying on existing signal detection algorithms. Reference [10] proposes OAMP-Net, a deep learning-enhanced version of the Orthogonal Approximate Message Passing (OAMP) iterative algorithm, aiming to improve signal detection performance by introducing trainable parameters on top of the original algorithm. Originally developed in the field of compressed sensing to solve sparse linear inverse problems, the OAMP algorithm was later applied to MIMO signal detection, offering significantly reduced complexity compared to previous methods. However, this reduction in complexity comes at the cost of decreased detection performance. OAMP is an iterative algorithm, which inherently increases computational complexity. To further reduce complexity, OAMP-Net consists of T cascaded layers, each corresponding to one iteration of the algorithm. Each layer not only implements the full OAMP algorithm but also incorporates trainable parameters, making the algorithm more flexible. These parameters allow OAMP-Net to adapt to a wider range of channel scenarios and enable transformation between different algorithmic models. Simulation results show that OAMP-Net outperforms both the original OAMP algorithm and the more complex LMMSE-TISTA algorithm, achieving lower computational complexity while maintaining adaptability to time-varying channels.

It is evident that the deep learning network proposed in Reference [8] requires the collection of a large amount of training data, resulting in a substantial initial training workload. In contrast, References [9] and [10] overcome this challenge by employing simpler training networks that achieve better signal detection performance.

Channel State Information (CSI) Feedback and Reconstruction

In frequency-division multiplexing (FDM) networks, the base station in MIMO systems needs to obtain downlink CSI feedback to perform pre-coding and achieve performance gains. However, the presence of massive antennas in MIMO systems leads to excessive feedback overhead, making traditional methods for reducing CSI feedback load unsuitable for such scenarios. Reference [11] proposes CsiNet, a CNN-based CSI sensing and recovery framework. The sensing part of CsiNet, also called the encoder, transforms the original CSI matrix into a codebook using a CNN. The recovery part, or decoder, reconstructs the original CSI signal from the received codebook using fully connected layers and CNNs. The encoder network consists of a 32×32 input layer, two 3×3 convolutional layers, a 1×N reshape layer, and a linear 1×M fully connected layer. The decoder network includes a 1×M input layer, a 1×N fully connected layer, a 32×32 reshape layer, and two Refine networks. Each Refine network contains four 3×3 convolutional layers for feature extraction. Reference [12] addresses CSI compression in spatial multiplexing MIMO systems by using a DNN to compress the original CSI matrix into a low-dimensional CSI signal but does not involve further recovery of the CSI feedback signal.

Building upon Reference [11], Reference [13] proposes a real-time CSI feedback framework called CsiNet-LSTM. This network employs CNNs and RNNs to extract spatial features and intra-frame temporal correlations of the CSI, respectively, thereby further improving the accuracy of CSI feedback. CsiNet-LSTM uses a time delay step of length T. At the first time step, the channel matrix is encoded using a high-compression-rate encoder, while the remaining T-1 time steps use low-compression-rate encoders. The output code words of the T-1 low-compression-rate encoders are

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 07, NCIRTAI - 2026

concatenated with the high-compression-rate encoder's output code word and then fed into the corresponding decoder. The final CSI reconstruction is performed by an LSTM with T time steps, each containing 3 layers of 2×32×32 units. It is noteworthy that the encoder and decoder architectures in this network are identical to those in the CsiNet structure of Reference [11]. By leveraging the temporal correlation and structural characteristics of time-varying MIMO channels, CsiNet-LSTM achieves a balance between compression ratio, CSI reconstruction quality, and computational complexity. Compared to CsiNet, this network trades time efficiency for improved CSI reconstruction quality. The CSI feedback and reconstruction algorithms proposed in References [11] and [13] both rely on large amounts of data for offline training. These networks have high complexity, and their generalization performance requires further investigation.

CHANNEL DECODING

Reference [14] proposes a deep neural network (DNN)- based channel decoding method. The study draws two key conclusions about applying deep learning to channel decoding: First, structured codes such as polar codes are easier to learn than random codes; second, for structured codes, deep learning networks can decode code words that were not seen during training. At the receiver, k information bits are encoded into a code word of length N, then modulated and transmitted through a noisy channel. The task of the channel decoder at the receiver is to recover the corresponding information bits from the noisy received code word. The channel decoder consists of an input layer, three hidden layers, and an output layer. The input layer receives the noise-corrupted code word, while the output layer produces the information bits. The three hidden layers have neuron structures of 128, 64, and 32 units, respectively. Deep learning-based channel decoding is inherently limited by the curse of dimensionality. For example, with a code length of 100 and a code rate of 0.5, there are 2^50 distinct code words, making this approach suitable only for channel coding with relatively short code words. Simulation results show that for structured codes, training 2^19 times approaches the performance of the maximum a posteriori (MAP) decoder, whereas for random codes, even after 2^19 training iterations, performance remains far below that of the MAP decoder. Additionally, Reference [14] compares different hidden layer configurations—128-64-32, 256-128-64, 512-256-128, and 1024-512-128—and finds that for this decoding network, more complex hidden layer structures require more training but yield better decoding performance. Reference [14] treats the decoding process as a black box, directly mapping received code words to information bits. Although this approach achieves performance comparable to traditional methods, the number of training iterations grows exponentially, and the deep learning network structure is sufficiently complex. When the code length changes, the network must be reconfigured for new input and output dimensions and retrained, resulting in a significant workload. Furthermore, this method is not suitable for random codes nor for long code words, thus having considerable limitations. In contrast, Reference [15] proposes a deep learning-based polar code decoding network with a separated sub-block structure, building upon traditional iterative polar code decoding algorithms. The network consists of two main steps: first, the original encoding/decoding is divided into M sub-blocks, each encoded/decoded independently, where the decoding of each sub-block employs a DNN with performance close to the MAP decoder. Introducing sub-blocks addresses the complexity problem caused by long code lengths. Second, the sub-blocks are connected via a belief propagation (BP) decoding algorithm. The BP algorithm and sub-block DNNs are connected to enable parallel processing. The decoding algorithm in Reference [15] is highly parallel and non-iterative. Compared to traditional algorithms, it significantly reduces decoding latency while maintaining comparable performance. Compared to the decoding algorithm in Reference [14], it greatly reduces both the number of training iterations and the complexity of the network structure.

Both references [16] and [15] combine the belief propagation (BP) algorithm with deep learning networks for channel decoding. Reference [16] proposes an iterative channel decoding algorithm called BP-CNN, which concatenates a CNN with a standard BP decoder to estimate information bits in noisy environments. At the receiver, the received signal is first processed by the BP decoder to obtain an initial decoded result. Then, the estimated transmitted symbols are subtracted from the received signal to obtain an estimate of the channel noise. Due to decoding errors, this noise estimate contains significant errors. Finally, the channel noise estimate is fed into the CNN to remove the estimation errors of the BP decoder. In this framework, the standard BP receiver estimates the transmitted signal, while the CNN reduces the estimation error of the BP detector and achieves a more accurate channel noise estimate. The iterative interaction between the BP algorithm and the CNN gradually improves the detection signal-to-noise ratio (SNR), resulting in better decoding performance. BP-CNN not only outperforms the standard BP algorithm in decoding performance but also has lower complexity, owing to the efficiency of CNN operations, which are mostly linear with only a small fraction being nonlinear. Simulation results show that the stronger the noise correlation, the greater the performance advantage of BP-CNN. When noise is uncorrelated, BP-CNN performs slightly worse than the BP algorithm.

High-density parity-check (HDPC) codes, the performance of the BP algorithm is relatively poor. The Tanner graph represents the parity-check matrix of the code. Nachmani et al. [17] proposed the BP-DNN algorithm, which assigns weighting coefficients to the edges of the Tanner graph and uses a deep neural network (DNN) to train and optimize these weights, thereby improving the performance of the BP algorithm when applied to HDPC codes. The BP-DNN requires approximately one-tenth the number of iterations of the standard BP algorithm while achieving better performance. Nachmani et al. [18] further proposed the BP-RNN algorithm, which combines recurrent neural networks (RNN) with the BP algorithm to further enhance its performance. Additionally, Reference [18] replaces the BP algorithm in BP-RNN with the modified random redundant iterative algorithm (mRRD), resulting in decoding performance superior to that of the original mRRD algorithm.

Among the aforementioned deep learning-based channel decoding methods, Reference [14] treats the decoding module in wireless communication systems as a black box. In contrast, other references combine deep learning with the belief propagation (BP) algorithm to further improve performance and reduce complexity.

End-to-End Wireless Communication Systems

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 07, NCIRTAI - 2026

O'Shea et al. [12] proposed a deep learning-based auto encoder physical layer scheme for MIMO systems. Under specific channel conditions, this scheme utilizes an auto encoder to globally optimize the processes of channel estimation, feedback, encoding, and decoding, aiming to maximize throughput and minimize bit error rate. Reference
[12] implemented three wireless communication systems using auto encoders: a spatial diversity MIMO system without CSI feedback, a spatial multiplexing MIMO system with perfect CSI feedback, and a spatial multiplexing MIMO system with compressed CSI feedback. Under certain channel environments, this physical layer scheme achieves significant performance improvements.

Reference [19] proposed a point-to-point wireless communication system model that demonstrates the feasibility of replacing the physical layer processing modules with deep neural networks (DNNs). Traditional wireless communication system design must account for various uncertainties in hardware implementation and perform compensations for delay, phase, and other factors. The DNN- based end-to-end wireless communication system proposed in [19] also considers these factors and conducts training in two stages. The first stage involves training the transmitter, channel, and receiver DNNs under a randomized channel model. In the second stage, based on the network parameters trained in the first stage, fine-tuning is performed under real channel conditions to further improve system performance. The channel module explicitly incorporates delay and phase compensation within the DNN training process. In the receiver module, feature extraction and phase compensation of the received signals are replaced by DNNs, with the outputs of two separately trained DNNs concatenated and fed into the receiver DNN. This DNN-based wireless communication system fully accounts for channel time variability under realistic conditions, achieving performance comparable to that of traditional wireless communication systems.

Reference [20] implements an end-to-end wireless communication system using deep neural networks (DNNs), where all signal-related functional modules—such as encoding, decoding, modulation, and equalization—are realized by DNNs. Instantaneous CSI in wireless communication systems is difficult to obtain accurately and continuously changes with time and location. Since the channel is unknown, gradient computation via back propagation in the end-to-end system cannot be directly performed. Reference [20] proposes an end-to-end system under unknown channel conditions that does not rely on any prior channel knowledge. The system employs a generative adversarial network (GAN) to model the wireless channel effects, using the encoded signal at the transmitter as conditional input. To overcome channel time variability, the received pilot signals are also included as part of the conditional information. In this wireless communication system, the transmitter and receiver are each replaced by a DNN, with the GAN serving as a bridge between them to enable smooth back propagation. The transmitter DNN, receiver DNN, and channel-generating GAN are trained iteratively, ultimately achieving a global optimum. Simulation results demonstrate that the GAN-based channel estimation method performs comparably to traditional channel estimation techniques, and the end-to-end wireless communication system achieves performance close to conventional communication systems based on expert channel models. This approach breaks the traditional model- driven wireless communication paradigm and opens a new path for wireless communication system design.

End-to-end wireless communication systems, also known as auto encoders, replace the traditional wireless communication system architecture with encoding, channel, and decoding processes—all implemented by deep learning networks. This represents a novel approach to wireless communication system design. However, multiple DNNs require a large amount of data, and the data collection process is highly demanding. Moreover, if the environment or hardware system changes, the data collection process must be repeated. Consequently, the practicality of implementing wireless communication systems using this approach remains limited at present.

Conclusions and future work

5G technology is characterized by high dimensionality, high capacity, and high density, generating massive amounts of data in wireless transmission. Big data at the physical layer has become a focal point of interest, with the aim of leveraging artificial intelligence to enhance transmission performance. In recent years, researchers have conducted preliminary explorations in this area, mainly presenting two types of deep learning networks: one driven purely by data and the other driven by both data and models. The data-driven deep learning networks [8, 11, 14, 19–20] treat multiple functional blocks of the wireless communication system as an unknown black box, replacing them with deep learning networks that rely on extensive training data to learn the mapping from input to output. For example, reference [8] treats the entire receiver module in an OFDM system as a black box. After the RF receiver obtains the signal and removes the cyclic prefix, a DNN is directly used to complete the process from the RF receiver to the destination. End-to- end wireless communication systems fully replace the entire communication system with deep learning networks, aiming to globally optimize the wireless communication system and achieve better performance [12, 20]. In contrast, data-and- model-driven deep learning networks [5, 7, 9–10, 15, 5,7,9–10,15,18] build upon the existing techniques of wireless communication systems without altering the system's model structure. They use deep learning networks to replace specific modules or to train related parameters, thereby enhancing the performance of those modules. For example, reference [6] leverages deep learning networks to introduce trainable parameters into the OAMP receiver for MIMO signal detection, further improving the performance of this module. While data-driven deep learning networks primarily rely on massive amounts of data, data-and-model- driven networks mainly depend on communication or algorithmic models.

Data-driven deep learning networks learn from a large number of examples, absorbing vast amounts of data that have been individually labeled by human analysts to generate the desired outputs. However, training such deep learning networks requires a substantial amount of labeled data, and the process of collecting and annotating this data is both time- consuming and costly. Beyond the challenge of data collection and labeling, most data-driven deep learning models exhibit weak generalization and adaptability; even minor changes in the network architecture can significantly degrade the accuracy of the trained model. For example, if the modulation scheme at the transmitter in reference [8] is changed to 16QAM (quadrature amplitude modulation) or

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 07, NCIRTAI - 2026

64QAM, the network needs to be retrained. Therefore, adjusting or modifying the model incurs a cost equivalent to recreating the model from scratch. To reduce the cost and time of training and tuning deep learning models, model- driven deep learning networks offer better generalization and adaptability. From 1G to 5G in cellular mobile communications, improvements in wireless system performance rely heavily on functional module modeling. Data-driven deep learning networks, which discard existing wireless communication knowledge, require massive datasets for training and learning, yet their achieved performance often falls short of traditional wireless communication system models. In contrast, model-driven deep learning networks, built upon existing physical layer models, can significantly reduce the amount of information needed for training or upgrading. Because existing models possess environmental adaptability and generalization capabilities, data-model- driven deep learning networks inherit these traits while further enhancing system performance based on the original models. A comparison between data-driven and data-model-driven deep learning networks is provided in Table 2. As analyzed in Section 3 of this paper, data-model-driven deep learning networks have demonstrated excellent performance in channel estimation, signal detection, and channel decoding, showing broad prospects for future development.

Table 2. Comparison of Deep Learning Networks: Data-Driven vs.
Data-Model Dual-Driven

Type	Data Dependency	Model Dependency	Accuracy	Complexity
Data-Driven	High	Low	Relatively Low	High
Data-Model Dual-Driven	Low	High	High	Low

CONCLUSION

The paper first introduces several widely used types of deep learning networks, including DNN, CNN, RNN, and GAN. It then elaborates on the latest research achievements of deep learning applied to wireless transmission technologies, covering channel estimation, signal detection, CSI feedback and reconstruction, channel decoding, and end-to-end wireless communication systems. As one of the mainstream technologies for post-5G development, intelligent wireless communication seeks breakthroughs at the physical layer by leveraging big data to reduce system implementation complexity and enhance system performance. The latest research progress indicates that data-model dual-driven deep learning networks not only meet these requirements but also significantly reduce dependency on data, making them one of the most promising directions for future development. Future enhancements include integrating AI with 6G networks, deploying edge AI for low latency, developing energy-efficient and lightweight models, enabling self- optimizing networks, improving spectrum management, strengthening AI-based security, and supporting massive IoT connectivity.

CONFLICTS OF INTEREST

The authors declare that they have no conflicts of interest.

ACKNOWLEDGMENT

I would like to express my special thanks of gratitude to my college management. I would like to give my thanks to guide who gave me the golden opportunity to do this wonderful project on Emerging Advances in Artificial Intelligence for Future Wireless Networks. In addition, I would also like to thank my parents who helped me a lot in finalizing this project within the limited time frame.

REFERENCES

[1]. NI S J, ZHAO J H. Key technologies in physical layer of 5G wireless communications network[J]. Telecommunications Science,2015, 31(12): 40-45.
[2]. MAO Q, HU F, HAO Q. Deep learning for intelligent wireless networks: a comprehensivey [J]. IEEE Communications Surveys & Tutorials, 2018(99):1.
[3]. O'SHEA T J, HOYDIS J. An introduction to deep learning for the physicalr [J]. arXiv: 1702.008320, 2017.
[4]. WANG T Q, WEN C K, WANG H, et al. Deep learning for wireless physical layer: opportunities and challenges [J]. China Communications, 2017, 14(11): 92-111.
[5]. HE H, WEN C K, JIN S, et al. Deep learning-based channel estimation for beamspace mmWave massive MIMOs [J]. IEEE Wireless Communications Letters, 2018(99): 1.
[6]. METZLER C A, MOUSAVI A, BARANIUK R G. Learned D-AMP: principled neural network based compressive image recovery[J]. arXiv: 1704.06625, 2017.
[7]. NEUMANN D, WIESE T, UTSCHICK W. Learning the MMSE channel estimator[J]. IEEE Transactions on Signal Processing,2018, 66(11): 2905-2917.
[8]. YE H, LI G Y, JUANG B H F. Power of deep learning for channel estimation and signal detection in OFDM systems[J]. IEEE Wireless Communications Letters, 2018, 7(1): 114-117.
[9]. SAMUEL N, DISKIN T, WIESEL A. Deep MIMO detection[C]//IEEE International Workshop on Signal Processing Advances in Wireless Communications, Jul 3-6, 2017, Sapporo, Japan. Piscataway: IEEE Press, 2017.
[10]. HE H, WEN C K, JIN S, et al. A model-driven deep learning network for MIMO detection[C]//Submitted to the 6th IEEE Global Conference on Signal and Information Processing, Nov 26-29, 2018, Anaheim, USA. Piscataway: IEEE Press, 2018.
[11]. WEN C K, SHIH W T, JIN S. Deep learning for massive MIMO CSI feedback[J]. IEEE Wireless Communications Letters,2018(99).
[12]. O'SHEA T J, ERPEK T, CLANCY T C. Deep learning based MIMO communications[J]. arXiv:1707.07980, 2017.
[13]. WANG T Q, WEN C K, JIN S, et al. Deep learning-based CSI feedback approach for time-varying massive MIMO channels[J]. arXiv:1807.11673, 2018.
[14]. CAMMERER S, HOYDIS J, BRINK S T. On deep learning-based channel decoding[C]//51st Annual Conference on Information Sciences and Systems, March 22-24, 2017, Baltimore, MD, USA. [S.l.:s.n.], 2017.
[15]. CAMMERER S, HOYDIS J, BRINK S T. Scaling deep learning-based decoding of polar codes via partitioning[J]. arXiv:1702.06901, 2017.
[16]. LIANG F, SHEN C, WU F. An iterative BP-CNN architecture for channel decoding[J]. IEEE Journal of Selected Topics in Signal Processing, 2018, 12(1): 144-159.
[17]. NACHMANI E, BEERY Y, BURSHTEIN D. Learning to decode linear codes using deep learning[C]//54th Annual Allerton Conference on Communication, Control, and Computing, Sept 27-31, 2016, Monticello, Illinois, USA. [S.l.s.n.], 2016.
[18]. NACHMANI E, MARCIANO E, LUGOSCH L, et al. Deep learning methods for improved decoding of linear codes[J].IEEE Journal of Selected Topics in Signal Processing, 2018,12(1): 119-131
[19]. .DÖRNER S, CAMMERER S, HOYDIS J, et al. Deep learning based communication over the air[J]. IEEE Journal of Selected Topics in Signal Processing, 2018, 12(1): 132-143.
[20]. YE H, LI G Y, JUANG B H F, et al. Channel agnostic end- to- end learning based communication systems with conditional GAN[J]. arXiv: 1807.00447, 2018.