Radar-Aided Beam Selection in MIMO Communication Systems: A Federated Transfer Learning Approach

Quan Zhou, *Student Member*, IEEE, Yongkang Gong, *Student Member*, IEEE, Arumugam Nallanathan, *Fellow*, IEEE, and Xianbin Wang, *Fellow*, IEEE

Abstract—By leveraging massive available data and hidden communication patterns, deep learning (DL) has enabled diverse applications in wireless networks operations. In this paper, we consider radar-aided beam prediction in multi-input multi-output (MIMO) communication systems with federated transfer learning (FTL) to preserve users' location privacy. Specifically, we propose a novel structure, i.e., radar-aided federated transfer beam prediction (RaFT-BP), to achieve few samples-enabled distributed beam selection in internet of vehicles (IoV) scenarios. Simulation results show that the proposed RaFT-BP can achieve the 93.78% top-5 accuracy with 600 samples in the distributed node, enabling 11.9% to 33.2% beam selection accuracy improvement compared with baseline schemes.

Index Terms—MIMO communications, beam prediction, federated transfer learning, internet of vehicles.

I. INTRODUCTION

Recently, the sixth generation (6G) wireless systems have attracted huge interest from both academia and industry, which can enable new services such as space-air-ground integrated networks, ubiquitous connectivity and smart cities [1], [2]. To support these emerging services, it is pivotal for 6G networks to provide high speed and dependable data transmission. Among many related technologies, MIMO serves as the foundation 6G networks by improving spectrum utilization efficiency in spatial domain with beamformed transmission. Beam selection can enhance MIMO systems' capacity and signal to noise ratio (SNR) by directing or dispersing the signal in a specific direction, which can be achieved via adjusting the relative position and phase of antennas.

There have been massive beam selection works. For example, Taranto *et al.* [3] utilized location information from both the base station and the user terminal to select the optimal beam index from a codebook. Pan *et al.* [4] established a mapping relationship between the receiver's position and the beam with the highest SNR at the receiver end. Additionally, Lin *et al.* [5] proposed a beam tracking algorithm that enhances tracking accuracy. However, these methods are implemented in ideal scenarios and do not fully leverage environmental information.

With increasing diversity and intelligent devices, different environmental data can be utilized for beamforming, which can improve communication efficiency due to decreasing overhead. By leveraging radar sensing information, radar-aided communication aims to enhance the performance of wireless communication [6], such as establishing secure and reliable communication links in dynamic environments. Moreover, position-aided [7] and LiDAR-aided methods [8] have been verified in real-world vehicle communication scenarios [9], [10].

While environmental information can improve communication performance, collecting location and motion may violate privacy

regulations [11]. To address this concern, McMahan *et al.* [12] proposed a federated learning (FL) architecture that adapts to communication scenarios. Elbir *et al.* [13] implemented a gradient information policy to update the beamforming model. To reduce beam search overhead, [14] considered that each user could train a shared FL model collaboratively for the distributed scenario. Zhang *et al.* [15] investigated backdoor attacks in FL beam selection systems and proposed corresponding defence strategies. However, the above researches ignore the limited computation power of nodes, which cannot match complex training models.

In this paper, we propose a radar-aided federated transfer learning (FTL) architecture for implementing beam prediction in MIMO communication systems, i.e., radar-aided federated transfer beam prediction (RaFT-BP). By transferring model weights instead of directly transmitting data, data privacy risks are eliminated and security is strengthened. Specifically, the pre-trained model in the edge computing unit (CU) is transmitted to the antenna computing units (ACUs) with few samples, which can reduce data dependency and computational energy consumption, conducting real-time training and dynamic prediction. The contributions of this paper can be summarized as follows.

- We propose a novel radar-aided federated transfer learning network for beam prediction under MIMO communication scenarios. Without local data sharing from the corresponding radars, the model weights of each local node are updated to achieve model migration.
- To reduce the model training overhead at each local node, the DL-based model is pre-trained in an edge CU with adequate data and the model weight is transmitted to ACU with few samples. Then, transfer learning (TL) is implemented in each ACU to reduce data reliance.
- Experimental results show that the proposed FTL-based algorithm performs better than the centralized approach in terms of beam prediction accuracy given imbalanced data. Additionally, we demonstrate that freezing the first five layers of neural networks yields optimal beam selection accuracy.

II. SYSTEM MODEL

A. Signal Model

1) Radar Signals: Assuming that the radar signal at the receiver side FMCW radar by transmitting a linear chirp signal starting from the initial frequency f_c and ramping up linearly to $f_c + \mu t$. Through a Delay Spread Fading channel $h(t) = Ae^{j(\phi+2\pi\Delta ft)} \cdot p(t-\tau)$ with amplitude gain A and time delay fading function $p(t-\tau)$, and the received signal is formulated as

$$R_r(t) = A^c A^t e^{\pi j(2\mu\tau t + 2f_c\tau - \mu\tau^2)}, \quad (1)$$

where A^c and A^t denote channel and transmitter gains.

The received radar signal is sampled at the sampling rate of f_n, generating N samples per period. For a FMCW radar with M receive antennas, a frame with B chirps of raw data can be expressed as $M \times N \times B$, which contains distance, angle, and velocity of moving objects. By applying the Fast Fourier Transform (FFT) on the time samples, the linear frequency-modulated signal is transformed into the frequency domain. Within this domain,

Quan Zhou is with the School of Electronics and Information Engineering, Beihang University. (e-mail: quanzhou@bupt.edu.cn).

Yongkang Gong is with the School of Computer Science and Technology, Shandong University, Qingdao, 266237, China (e-mail: gokawa@sdu.edu.cn)

Arumugam Nallanathan is with Queen Mary University of London. (e-mail: a.nallanathan@qmul.ac.uk).

Xianbin Wang is with Department of Electrical and Computer Engineering, Western University, London, ON N6A 5B9, Canada (e-mail: xianbin.wang@uwo.ca).
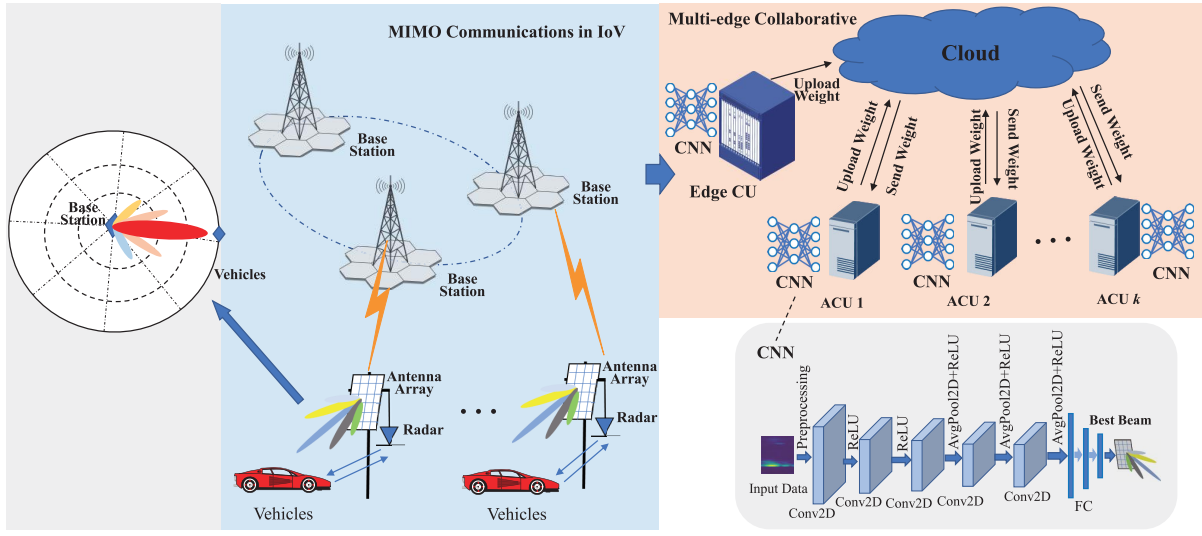
Fig. 1. An illustration of the proposed RaFT-BP, which includes an edge CU with adequate data and various ACUs with few samples.

the linear frequency-modulated signal is shifted proportionally to the round-trip duration, generating distance feature. Then, more distinctive features are obtained by averaging out chirp samples. The angle feature is obtained by applying FFT on angular samples. Accordingly, the range-angle feature of radar signals can be regarded as the primary factor for tracking vehicles, which are calculated as

$$R_a = \sum_{b=1}^{B} \int \int R_r(m,n,b)e^{-2\pi j(mu+nv)}dmdn, \qquad (2)$$

where $R_a \in \mathcal{C}^{M \times N}$ is the range-angle feature.

2) Communication Signals: The mmWave communication signal at the transmitter end propagates through channel **H**, which is denoted as

$$\mathbf{H} = \sqrt{\frac{N_t N_r}{N_c N_{ray}}} \sum_{i=1}^{N_c} \sum_{j=1}^{N_{ray}} \gamma \beta_t(i,j)\beta_r(i,j), \qquad (3)$$

where N_t denotes the number of transmitter antennas, N_r signifies the number of receiver antennas, N_c is the scattering cluster, and N_{ray} indicates the number of propagation paths; γ is the decay factor; β_t presents the normalized antenna array response vector in the transmitter, and β_r in receiver. Consequently, the received MIMO signal at the receiver end can be expressed as

$$R_s(t) = A\mathbf{H}s(t,b) + n(t), \qquad (4)$$

where A denotes the wireless channel gain; $s(t,b)$ presents the mmWave signal, and b is the received beam; $n(t)$ signifies the Additive White Gaussian Noise (AWGN). In this paper, the radar data R_r is processed to derive the feature R_a for the selection of the optimal radio beam b.

B. Problem Definition

The deployment of radar sensing at communication terminals can provide critical information about transmitter/receiver position and environment, which can reduce beam training overhead in mmWave and MIMO communication systems, enabling low-energy, low-latency applications. In real-world scenarios, various radar sensing nodes are distributed in various parts of the city block, each sensing node needs to have the ability to model training, recognition, and processing. There can be significant differences between nodes, such as varying amounts of training data for each node and hardware limitations that prevent large-scale training. Meanwhile, to ensure security, data islands exist between various nodes, and the risk of leakage exists in direct

sensing data transmission. Besides, the few samples processed by each node are prone to undesirable beam prediction accuracy.

Assuming that there are K ACUs with few samples and an edge CU with adequate data, we aim to design an architecture $\mathcal{M}$ to enable various nodes with few samples that implement distributed beam prediction. The process can be expressed as

$$\mathcal{F}_k \leftarrow \mathcal{M}(F|E_l, A_i), \qquad (5)$$

$$b \leftarrow \mathcal{F}_k(R_a), \qquad (6)$$

where E_l denotes the hardware with weak computing capacity. A_i indicates the data isolation environment. F represents the untrained CNN model, and $\mathcal{F}_k$ signifies the trained model in the ACU k, which can implement beam prediction; b is the optimal beam in ACU k.

C. Proposed RaFT-BP Framework

Fig. 1 presents a novel distributed approach for radar-assisted beam prediction, where the ML-based model is designed to predict the optimal beam index from a predefined beam index using radar data captured by a base station. The ML model aims to maximize the received SNR for wireless communications by returning the beam index from the code book with the highest probability of achieving the design. Given the data isolation in distributed ACUs and the risk of data eavesdropping in the channel, we propose a method based on FTL to transfer model weights between nodes to mitigate these issues. As shown in the multi-edge collaboration part of Fig. 1, the multiple ACUs involved in collaborative training are regarded as different nodes, and the cloud server controls the interaction between CU and various ACUs. The proposed architecture reduces the computational complexity and sample dependency in distributed beam selection by means of FTL. Signalling interactions between various base stations and the core network in the communication network are not the main subject of this work. Accordingly, the communication between various base stations and the core network is virtualized into links, mainly considering the transmission of training weights between individual computing nodes and the cloud server.

Assuming the raw radar measurement data denoted by $R = \{r_{es}, r'_{s_1}, r'_{s_2}, \ldots, r'_{s_K}\}$, which is limited across each node due to data isolation in various ACUs. To address this limitation, we propose a solution that involves training a deep neural network (DNN) model at the edge CU. This DNN model can be transferred to meet beam prediction requirements of each ACU, reducing the need for large amounts of training data. In this approach, we

consider the source domain data at the edge CU, denoted as $r_{es} = \{(r_i^a, y_i^a)\}_{i=1}^{N_a}$, and the client node data as $r'_{s_k} = \{(x_j^b, y_j^b)\}_{j=1}^{N_b}$.

As shown in Fig. 1, the DNN-based model incorporates a multi-layer convolutional module that reduces the complexity of the data and extracts key features. The model's weights, trained on the edge CU's data (source domain), are then transmitted to the various electromagnetic environments where each ACU is located (target domain). This approach alleviates the data dependence of the DL model in ACU with limited samples by freezing specific parameters (no re-training).

The process involves training the DNN model on edge CU and transferring model weights to each sensing node. Fig. 2 depicts the details of the fine-tuning of each node, and the details of the CNN-based model are described in Table I. To solve complex wireless environments, we utilized the appropriate AvgPooling and resized convolutional kernels in the structure design. In detail, the DNN model contains five convolutional layers and three fully connected layers. Since radar image features have a large background noise, the initial two layers do not use AvgPooling operations, so channel noise and interference may persist. Filters are increased from 8 to 16 and then gradually decreased to 2. As the network gets deeper, gradually increasing the number of filters enables the network to learn more diverse and complex features from the raw radar signal. Then, decreasing the number of filters reduces computational complexity. Excessive filters may require more computational resources. In comparison, a few filters can reduce the overfitting risk, and speed up convergence.

For the proposed mechanism, the structure of the model on both CU and various ACUs is same. First, the CU implements pre-training. Based on the pre-trained model received from CU, TL is implemented on each ACU, fixing some parameters and retraining some of them. The process of edge CU pre-training and transferring to various ACUs can be formulated as

$$
\begin{cases}
Edge\ CU : \mathcal{F}_p \leftarrow \mathcal{M}[F|r_{es}], \\
ACU\ 1 : \mathcal{F}_1 \leftarrow \mathcal{M}[F|\mathcal{F}_p, l, r'_{s_1}], \\
ACU\ 2 : \mathcal{F}_2 \leftarrow \mathcal{M}[F|\mathcal{F}_p, l, r'_{s_2}], \\
\quad \cdots \\
ACU\ K : \mathcal{F}_K \leftarrow \mathcal{M}[F|\mathcal{F}_p, l, r'_{s_K}],
\end{cases}
\tag{7}
$$

where $\mathcal{M}$ is the proposed FTL architecture for beam prediction. F denotes the untrained CNN model, $\mathcal{F}_p$ represents the edge CU, and $\mathcal{F}_k$ signifies the trained model in the ACU k. Then, each ACU uploads the model parameters to the cloud server and performs averaging aggregation to yield aggregated model $F_a = 1/k \sum_K F_k$. Next, F_a is transmitted to each ACU, conducting TL. From aggregation of global models to TL is one iteration. Multiple iterations are performed until the model of each ACU converges.

As there is no direct data transfer and few sample requirements, FTL-based beam prediction offers a more secure and reliable approach for smart sensing applications. By allowing each ACU to map its unique feature space during the TL process, the requirement for all nodes to use the same feature data is eliminated. This ensures that each node can implement beam prediction based on its own data. However, since the data is available at each node, the DL-based model is susceptible to over-fitting. Additionally, directing transfer data to each ACU may result in communication overload and increase data leakage risk. To address these concerns, the proposed RaFT-BP delivers the pre-trained model between various nodes. The optimization objective is defined as

$$
\min\ loss(\mathcal{M}(F|\mathcal{F}_p, R), \mathcal{Y}),
\tag{8}
$$

where $\mathcal{F}_p$ signifies the pre-trained model from the edge CU; F denotes the transferred model in various ACU. $\mathcal{M}(\cdot)$ is the FL algorithm. $\mathcal{Y}$ represents the true label of R. $loss(\cdot)$ is the cross entropy function.
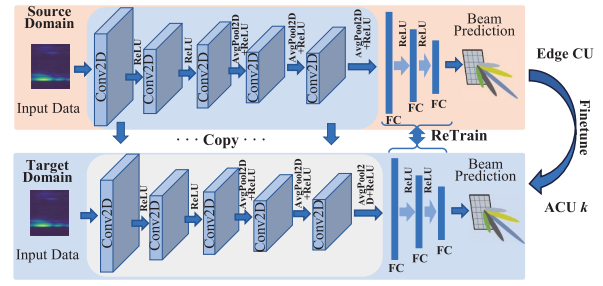

Fig. 2. The TL process of beam prediction for each CU node.

TABLE I
THE DETAILS OF CNN-BASED BEAM PREDICTION MODEL IN RAFT-BP

Module	Layer	Output
Input Layer	/	$1 \times 256 \times 64$
CNN Module	Conv2D(filters 8, size 3×3, padding 2)+Relu	$8 \times 256 \times 64$
	Conv2D(filters 16, size 3×3, padding 2)+Relu	$16 \times 256 \times 64$
	Conv2D(filters 8, size 3×3, padding 2)+AvgPool2d(size 2, stride 2)+Relu	$8 \times 128 \times 32$
	Conv2D(filters 4, size 3×3, padding 2)+AvgPool2d(size 2, stride 2)+Relu	$4 \times 64 \times 16$
	Conv2D(filters 2, size 3×3, padding 2)+AvgPool2d(size 2, stride 2)+Relu	$2 \times 32 \times 8$
Flatten	/	512
FC Module	FC+Relu	512×256
	FC+Relu	256×128
	FC	128×64

For the scheduling of the proposed architecture, the process of radar signal collection begins with pre-training on the Edge CU. This pre-trained model implements beam prediction based on Edge CU data. Once the data collected at ACUs reaches a certain threshold, the pre-trained model is fine-tuned based on the TL mechanism. In the dynamic environment of radar-assisted beamforming, the approach relies on the periodic transmission of radar signals by transmission units. Based on the codebook, the raw data of radar echos is analyzed by the trained model, which infers the optimal beam-index. The corresponding beamforming matrix, along with the phase adjustments of the antenna array, is then used to form the required beam direction.

D. Computational Complexity

Time complexity is used to represent the computational complexity of an algorithm, which measures computational efficiency and resource consumption. Floating point operations (FLOPs) can evaluate the time complexity of the model [16]. For the CNN structure in the proposed algorithm, it consists of five convolutional layers and three fully connected layers. The time complexity of convolutional layer and fully connected layer are formulated as

$$
T_c^w \sim O\left((F_{l-1}^h \cdot F_{l-1}^w) \cdot (H \cdot W) \cdot c_{l-1} \cdot c_l\right),
\tag{9}
$$

$$
T_c^f \sim O\left((F_{l-1}^h \cdot F_{l-1}^w) \cdot c_{l-1} \cdot c_l\right),
\tag{10}
$$

where T_c^w signifies the time complexity of convolutional layer and T_c^f represents the time complexity of fully connected layer; $H \cdot W$ denotes the kernel size; c_{l-1} and c_l are input and output channels of l-th layer; $(F_l^h \cdot F_l^w)$ represents the feature map of l-th layer. In this way, for the non-TL approach, the time complexity T_{NT} for k nodes is

$$
T_{NT} = k\left(a \cdot T_c^w + b \cdot T_c^f\right)
\tag{11}
$$

Settings	Assignments
Feature	Range-Angle
Data format	[1×256×64]
Number of data in edge CU	2000
Number of data for each ACU	ACU1: 400, ACU2: 500, ACU3: 400, ACU4: 400, ACU5: 300
Learning rate	Edge CU & ACUs: 0.01 (Reduced by half every 20 rounds)

Fig. 3. Beam selection accuracy and training loss values with number of iterations under original and SNR = {0, 5, 10} dB conditions.

where a and b are the number of convolutional layers and fully connected layers. For the TL approach, the time complexity T_{FTL} for k nodes is

$$T_{\text{FTL}} = \begin{cases} k\left(a \cdot T_c^w + (b-l) \cdot T_c^f\right), & 0 \le l \le b, \\ k\left((a-(l-b)) \cdot T_c^w\right), & l > b. \end{cases} \quad (12)$$

where l signifies the number of re-trained layers. Accordingly, the ratio of time complexity P_T is expressed as

$$P_T = \begin{cases} \left(1 - \dfrac{l \cdot T_c^f}{a \cdot T_c^w + b \cdot T_c^f}\right) \times 100\%, & 0 \le l \le b, \\ \left(\dfrac{(a+b-l) \cdot T_c^w}{a \cdot T_c^w + b \cdot T_c^f}\right) \times 100\%, & l > b. \end{cases} \quad (13)$$

Consequently, the gain from TL over the traditional FL approach is $(1 - P_T)$.

III. EXPERIMENTS AND DISCUSSIONS

A. Dataset and Settings

The proposed RaFT-BP scheme is evaluated by conducting experiments on the DeepSense Scenario 9 [17], where a 64-beam code is considered, the number of training data is 4199, and test data is 593. In this way, the beam prediction accuracy is defined as a 64-classification issue. Additionally, the Range-Angle feature is employed to conduct our experiments, which is computed by (2). Table II provides the details of the experimental dataset. Dynamic learning strategies are implemented in this experiment. In the early stages of model training, a larger learning rate can help the model escape from local optima and accelerate the gradient towards the global optimum. As the model approaches the optimal solution, a smaller learning rate can lead to a more stable convergence. The experiments are conducted using an NVIDIA GeForce GTX 1080Ti GPU, and the PyTorch framework implements the proposed scheme.

B. Prediction Performance

To evaluate the influence of wireless parameters on the convergence of the proposed technique, Additive White Gaussian Noise (AWGN) is added to the DeepSense dataset in order to examine the effect of SNR on experimental convergence. In this way, datasets with SNR= {0 dB, 5 dB, 10 dB} are obtained. Fig. 3 illustrates the change in accuracy and loss values during training. Experimental results illustrate the effect of wireless channel noise parameters on model convergence, where model convergence becomes slower and the upper limit of accuracy decreases in a lower SNR environment. After 20 rounds of training, accuracy and loss of training set converge gradually. According to the results, the proposed algorithm converges quickly in weak computational power and resource-constrained environments, and implements TL at various nodes in the federated architecture.

We present the performance of beam prediction in Fig. 4, where the proposed scheme is compared with [6] (Baseline), the No-FTL method, and the FTL approach. Specifically, the baseline algorithm considers an architecture trained on the data from each ACU. The No-FTL scheme represents a pre-trained model trained
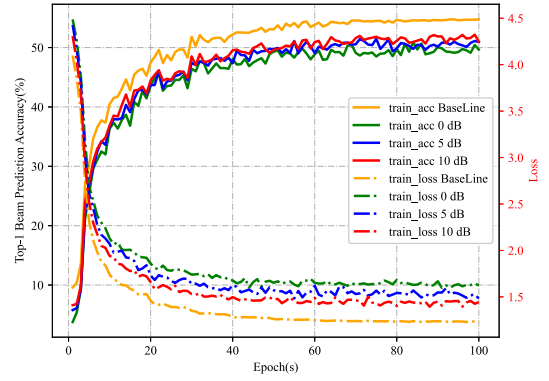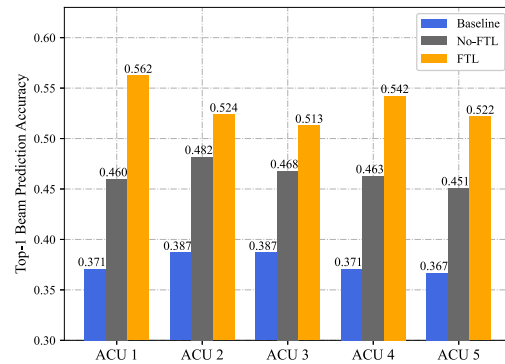

Fig. 4. The top-1 accuracy of three mechanism implementing beam prediction for each node.

on the cloud server, which conducts beam prediction on each ACU. The FTL approach implements TL based on the pre-trained model to perform beam prediction on various ACUs.

Fig. 4 highlights the impact of different training approaches on beam prediction accuracy. When DL models are trained solely on isolated data from each node, the beam selection accuracy for each ACU ranges from 36.7% to 38.7%. Whereas, by utilizing a pre-trained model from the edge CU, the beam selection accuracy improves from 45.1% to 48.2%. Furthermore, implementing TL at each ACU node further enhances the beam prediction accuracy, resulting in accuracy ranging from 51.3% to 56.2%. Compared with the baseline mechanism, the No-FTL scheme achieves a 7.3% to 11.5% improvement, while the FTL approach surpasses the baseline mechanism by 13.1% to 19.5%. Notably, the proposed approach effectively addresses the challenge of data isolation by transmitting model parameters. Directly applying No-FTL methods to each node, where the model is trained on a few samples, results in limited inference and recognition capabilities. Consequently, by implementing federated transfer learning, FTL achieves a significant improvement over No-FTL, which is achieved with much fewer weight parameters.

The performance of the model is measured using both top-1 accuracy and top-5 accuracy. Top-5 accuracy provides a better assessment of beam assignment fault tolerance, where multiple suitable beams may appear in a communication system rather than one absolutely correct answer. It still makes sense to use the model even if the first option is not the best beam and the correct beam is in the first five options.

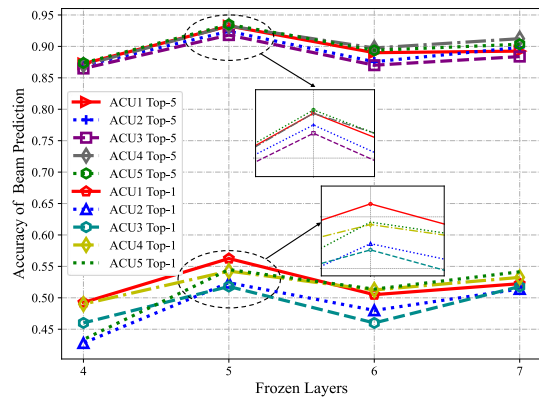Fig. 5 displays the comparison of top-1 and top-5 accuracy

Fig. 5. The top-1 and top-5 accuracy of beam prediction for each ACU under different frozen layers.

Fig. 6. The top-1 and top-5 accuracy of beam prediction for the proposed scheme and baseline methods under different training samples.

of beam prediction for each ACU under different frozen layers. We conducted TL in the experiments by freezing four, five, six, and seven layers. When four layers were frozen, the top-1 accuracy for beam prediction ranged from 42.8% to 49.3% across various nodes, while the top-5 accuracy varied from 86.48% to 87.36%. With five frozen layers, the top-1 accuracy ranged from 51.8% to 56.3%, and the top-5 accuracy was between 91.8% and 93.49%. For six frozen layers, the top-1 accuracy ranged from 46.0% to 51.4%, and the top-5 accuracy ranged from 87.01% to 89.75%. When seven layers were frozen, indicating retraining only the last layer of the fully connected layer, the maximum top-1 accuracy achieved for beam prediction was 54.1%, and the top-5 accuracy was 91.23%. These results demonstrate that the best beam prediction performance is obtained when five layers are frozen. On the other hand, freezing four layers led to poor accuracy due to insufficient training data for the convolutional part of the model. The findings emphasize the importance of selecting the appropriate number of frozen layers to achieve optimal beam prediction accuracy.

In Fig. 6, the proposed method is compared with several baseline approaches, including LeNet, FPN, VGGNet, ResNet, Transformer, LSTM, Look Table [7], and FLHB [13]. All algorithms are implemented under the conditions of 200, 300, 400, 500, and 600 training samples, and the beam prediction accuracy of each approach is shown in Fig. 6. Specifically, for 200 training samples, the top-1 accuracy of baseline methods varies from 21.4% to 24.6%, and the max top-5 accuracy of beam prediction reaches 51.61%. As the number of training samples increases, the accuracy of the baseline methods also increases accordingly. When the training sample equals 600, the accuracy of the baseline methods ranges from 36.4% to 39.9%, and the max top-5 accuracy of beam prediction is 60.88%. The top-1 beam prediction accuracy of the proposed mechanism exceeds 54.6%, and the top-5 reaches 93.78%, exceeding that of the baseline methods by 6.2% to 32.9%. These experimental results demonstrate that the proposed mechanism effectively addresses the issue of low beam prediction accuracy under distributed conditions.

IV. CONCLUSION

In this paper, we propose the RaFT-BP architecture, which aims to improve beam prediction accuracy under the constraints of few samples and distributed conditions. The effectiveness of the proposed RaFT-BP mechanism is verified on the DeepSense dataset. Experimental results demonstrate that the proposed mechanism achieves promising beam prediction accuracy, ranging from 51.8% to 93.78%. Moreover, numerical experiments reveal that freezing five layers yields superior beam prediction accuracy. The proposed algorithm outperforms baseline schemes with an average improvement ranging from 11.9% to 33.2%. These results demonstrate
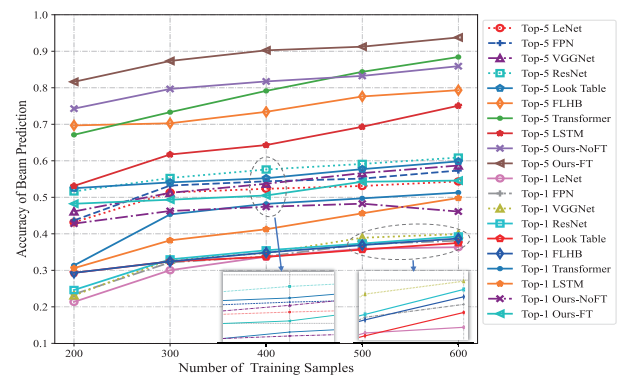
that the proposed RaFT-BP mechanism effectively addresses the challenge of low beam prediction accuracy in distributed settings with data isolation.

REFERENCES

[1] D. Serghiou, M. Khalily, T. W. C. Brown and R. Tafazolli, "Terahertz Channel Propagation Phenomena, Measurement Techniques and Modeling for 6G Wireless Communication Applications: A Survey, Open Challenges and Future Research Directions," *IEEE Commun. Surv. Tutor.*, vol. 24, no. 4, pp. 1957-1996, 2022.

[2] Y. Gong, H. Yao, J. Wang, M. Li, and S. Guo, "Edge intelligence-driven joint offloading and resource allocation for future 6G industrial Internet of Things," *IEEE Trans. Netw. Sci. Eng.*, Early Access, 2022.

[3] R. DiTaranto, S. Muppirisetty, R. Raulefs, D. Slock, T. Svensson and H. Wymeersch, "Location-Aware Communications for 5G Networks: How location information can improve scalability, latency, and robustness of 5G," *IEEE Signal Process. Mag.*, vol. 31, no. 6, pp. 102-112, Nov. 2014.

[4] A. Pan, T. Zhang and X. Han, "Location information aided beam allocation algorithm in mmWave massive MIMO systems," in *Proc. IEEE/CIC Int. Conf. Commun. China*, Qingdao, China, pp. 1-6, 2017.

[5] Y. Lin, C. Shen and Z. Zhong, "Sensor-Aided Predictive Beam Tracking for mmWave Phased Array Antennas," in *Proc. IEEE Glob. Commun. Conf., GLOBECOM Workshops*, Waikoloa, HI, USA, 2019, pp. 1-5.

[6] U. Demirhan and A. Alkhateeb, "Radar Aided 6G Beam Prediction: Deep Learning Algorithms and Real-World Demonstration," in *Proc. IEEE Wireless Commun. Networking Conf.*, Austin, USA, pp. 2655-2660, 2022.

[7] J. Morais and A. Alkhateeb, "Position-aided Beam Prediction in the Real World: How useful GPS positions actually are?", in *Proc. Int. Conf. Commun.*, Rome, Italy, pp. 1824-1829, 2023.

[8] S. Jiang, G. Charan and A. Alkhateeb, "LiDAR Aided Future Beam Prediction in Real-World Millimeter Wave V2I Communications," *IEEE Wirel. Commun. Lett.*, vol. 12, no. 2, pp. 212-216, Feb. 2023.

[9] G. Charan, T. Osman, A. Hredzak, N. Thawdar and A. Alkhateeb, "Vision-Position Multi-Modal Beam Prediction Using Real Millimeter Wave Datasets," in *Proc. IEEE Wireless Commun. Networking Conf. WCNC*, Austin, TX, USA, 2022, pp. 2727-2731.

[10] G. Charan et al., "Towards Real-World 6G Drone Communication: Position and Camera Aided Beam Prediction," in *Proc. IEEE Glob. Commun. Conf., GLOBECOM*, Rio de Janeiro, Brazil, pp. 2951-2956, 2022.

[11] S. Stalla-Bourdillon, H. Pearce, and N. Tsakalakis, "The GDPR: A game changer for electronic identification schemes? The case study of Gov. U.K. verify," *Comput. Law Secur. Rev.*, vol. 34, no. 4, pp.784-805, Aug. 2018.

[12] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. Y. A. Arcas, "Communication-efficient learning of deep networks from decentralized data,' in *Proc. Artif. Itell. Statist.*, pp.1273-1282, 2017.

[13] A. M. Elbir and S.Coleri,"Federated learning for hybrid beamformingin mmwave massive MIMO," *IEEE Commun. Lett.*, vol. 2, no.12, pp.2795-2799, Dec. 2020.

[14] M. B. Mashhadi, M. Jankowski, T.-Y. Tung, S. Kobus, and D. Gunduz, "Federated mmWave beam selection utilizing LIDAR data," *IEEE Wireless Commun. Lett.*, vol. 10, no.10, pp. 2269-2273, Oct. 2021.

[15] Z. Zhang, R. Yang, X. Zhang, C. Li, Y. Huang and L. Yang, "Backdoor Federated Learning-Based mmWave Beam Selection," *IEEE Trans. Commun.*, vol. 70, no. 10, pp. 6563-6578, Oct. 2022.

[16] K. He and J. Sun, "Convolutional neural networks at constrained time cost," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, pp. 5353–5360, 2015.

[17] A. Ahmed, G. Charan, T. Osman, A. Hredzak, J. Morais, U. Demirhan and N. Srinivas. "DeepSense 6G: A Large-Scale Real-World Multi-Modal Sensing and Communication Dataset", *IEEE Commun. Mag.*, vol. 61, no. 9, pp. 122-128, September 2023.