Agile Semantics Alignment over Fading Channel via LoRA-based Fine-Tuning

Sang-Hyeok Kim, Joonhoe Koo, and Seung-Woo Ko
Inha University, Incheon, Korea
Email: shkim9151@gmail.com, jeremy0915@inha.edu, and swko@inha.ac.kr

***Abstract*—This paper proposes a communication and learning framework to rapidly resolve *semantics misalignment* (SMA) between a TX and an RX in dynamic fading channels. The SMA arises from a mismatch in feature vector dimensions when wireless channel quality degrades. Our approach is based on a split-architecture semantic communication system, where the *transmitter* (TX) uses *principal component analysis* (PCA) for dimensionality reduction. The *receiver* (RX) then reconstructs the transmitted feature vector and performs fine-tuning on its pre-trained model using *Low-Rank Adaptation* (LoRA) to swiftly adapt to the changing channel conditions. Experimental results on a V2X testbed show that our method achieves significant latency reduction compared to end-to-end training, while maintaining stable and high accuracy compared to using a pre-trained model without fine-tuning.**

***Index Terms*—Semantic communication, fine-tuning, LoRA, fading channels, V2X.**

I. Introduction

Next-generation wireless networks will handle vast amounts of multimodal data from applications like the *Internet of Things* (IoT) and autonomous driving. Traditional bit-based communication faces a communication bottleneck due to limited bandwidth and channel variations, leading to information loss and latency [1]. To overcome this, *semantic communication* (SC), which focuses on conveying meaning rather than just bits, has emerged as a key technology for the 6G era [2].

However, the quality of wireless channels constantly varies, altering the feasible dimension of transmittable feature vectors. This leads to a *semantics misalignment* (SMA) between the dimension sent by the TX and the dimension expected by the pre-trained model at the RX, drastically degrading task success rates [3]. As visually demonstrated in Fig. 1, this problem causes severe degradation in reconstructed images when the encoder and decoder are trained under different channel conditions or datasets. Retraining the entire model for each dimensional change is infeasible due to high latency and resource requirements.

To address this challenge, this paper proposes a low-latency solution that maintains the integrity of the pre-trained model. We introduce a framework that combines feature dimension control at the TX with a rapid fine-tuning mechanism at the RX, ensuring resilient and efficient communication in dynamic channel environments.

II. System Model and Problem Definition

We consider a split-architecture SC system with a point-to-point link between a single TX and RX. An input image is

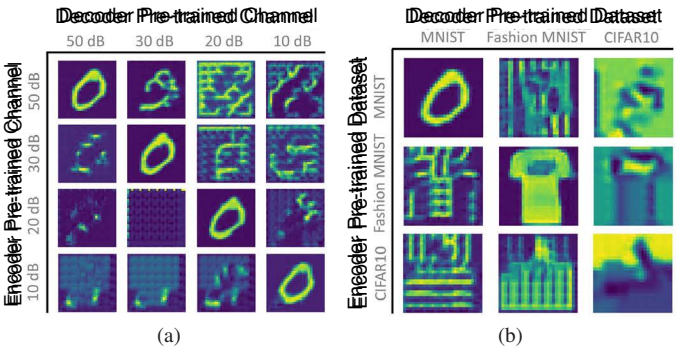

Fig. 1. Examples of image reconstruction results under SMA. (a) Mismatched channel SNRs. (b) Different training datasets. This misalignment occurs between the encoder and decoder [3].

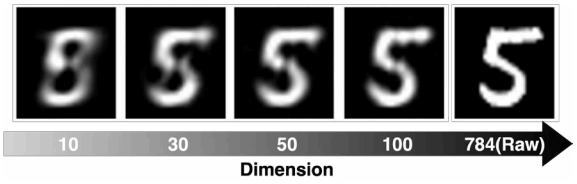

Fig. 2. Reconstructed images of an MNIST digit ('5') based on varying numbers of PCA dimensions. The degradation in quality at lower dimensions highlights the SMA challenge for a fixed RX model.

represented by a vector $\boldsymbol{x} \in \mathbb{R}^{d_{\text{raw}}}$, which corresponds to the dimension of the raw, uncompressed data (e.g., for a 28x28 MNIST image, $d_{\text{raw}} = 784$). To transmit this data efficiently, the TX encoder compresses $\boldsymbol{x}$ into a lower-dimensional feature vector $\boldsymbol{z} \in \mathbb{R}^{\alpha}$. The dimension α is adaptively controlled based on the real-time channel state. At the RX, the decoder reconstructs the data $\hat{\boldsymbol{x}} \in \mathbb{R}^{d_{\text{raw}}}$ from $\boldsymbol{z}$.

The subsequent classification model, with pre-trained weights $\boldsymbol{W}$, is designed to achieve high accuracy under good channel conditions. This pre-training is performed on data that has been reconstructed from an ideal, high-quality set of d_{pre} principal feaures, where $d_{\text{pre}} < d_{\text{raw}}$ (e.g., $d_{\text{pre}} = 200$ for the MNIST model in our experiments). The model is therefore optimized for the rich information content associated with this d_{pre} dimension. The core challenge arises when poor channel conditions force the transmission dimension α to be much smaller than the model's expected dimension d_{pre}, which degrades the quality of the reconstructed image as shown in Fig. 2.

The fluctuating quality of wireless channels forces a trade-off between two critical objectives, which defines the core problem of this paper:

- **Meeting Data Rate Requirements:** When channel quality degrades, transmitting the high-dimensional feature vector (d_{pre}) that the RX model is aligned with can violate latency and data rate constraints. To comply, the TX must reduce the transmission dimension to $\alpha < d_{\text{pre}}$.
- **Maintaining Task Performance:** This necessary reduction in dimension, however, leads directly to the SMA problem. As the information content of the reconstructed data $\hat{\boldsymbol{x}}$ deviates from what the pre-trained model $\boldsymbol{W}$ expects, the final task performance (e.g., classification accuracy) drops significantly.

Therefore, our goal is to resolve this conflict. We aim to develop a system that meets strict channel requirements by adjusting the transmitted data, while simultaneously preventing performance loss by adapting the receiver model.

III. Proposed Algorithm: Agile Semantics Alignment

To resolve the trade-off between data rate and task accuracy defined in Section II, we propose a novel framework for agile semantics alignment. The core idea is to treat the mandatory reduction of feature dimensions not just as a challenge, but as an opportunity. By significantly lowering the dimension of the transmitted data, we first meet the strict data rate requirements imposed by the channel. This reduction creates spare capacity in the communication payload, which we then leverage to transmit a small set of training data. At the RX, this data is used to rapidly fine-tune the pre-trained model, allowing it to adapt to the lower-dimensional input and thus recovering task performance. This co-adaptation of communication payload and the receiver model enables a system that is resilient to dynamic channel conditions.

A. Adaptive Payload Composition via PCA

To implement the dimensionality reduction and reconstruction processes described in the system model, our framework employs *Principal Component Analysis* (PCA) [4]. The TX uses a pre-calculated projection matrix $\boldsymbol{P}_\alpha \in \mathbb{R}^{d_{\text{raw}} \times \alpha}$, which contains the top α principal components as its columns. The compression (encoding) of an input vector $\boldsymbol{x}$ is performed as:

$$\boldsymbol{z} = \boldsymbol{P}_\alpha^\top \boldsymbol{x} \tag{1}$$

where $\boldsymbol{z} \in \mathbb{R}^\alpha$ is the compact feature vector transmitted over the channel. This reduction creates spare transmission capacity, which is then strategically used to send a small set of training samples that enable the RX to perform agile fine-tuning. The maximum number of transmittable training samples, $N_{\max}$, for a given maximum payload size $B_{\max}$ is calculated as:

$$N_{\max} = \left\lfloor \frac{B_{\max} - H - t(h + \sigma\alpha)}{h + \sigma\alpha} \right\rfloor \tag{2}$$

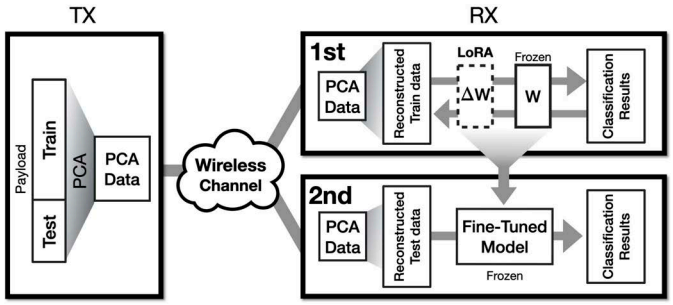

Fig. 3. The overall online framework of the proposed LoRA-based fine-tuning system for agile semantics alignment.

where H is the main header, h is the vector header, t is the number of test samples, and σ is the bytes per dimension value. At the RX, the data is reconstructed (decoded) via:

$$\hat{\boldsymbol{x}} = \boldsymbol{P}_\alpha \boldsymbol{z} \tag{3}$$

The reconstructed vector $\hat{\boldsymbol{x}}$, while having the same dimension as the original raw data, contains less information due to being reconstructed from the compressed vector $\boldsymbol{z}$. Feeding this information-degraded data into the pre-trained classifier causes severe performance degradation due to SMA. The following subsection details our proposed method to overcome this degradation.

B. LoRA-based Fine-Tuning

As established in Section III-A, the TX lowers the feature dimension via PCA to satisfy channel capacity constraints, which in turn causes SMA. To overcome the subsequent performance degradation, the RX performs a rapid, low-latency adaptation of its classification model using *Low-Rank Adaptation* (LoRA) [5]. LoRA is a parameter-efficient fine-tuning technique that avoids retraining the entire model. Instead, for a pre-trained weight matrix $\boldsymbol{W} \in \mathbb{R}^{d_{\text{raw}} \times k}$ within the classifier, where k is the number of classes, it injects a small, trainable update $\Delta\boldsymbol{W}$. This update is represented by a low-rank decomposition:

$$\Delta\boldsymbol{W} = \boldsymbol{B}\boldsymbol{A}, \tag{4}$$

where $\boldsymbol{B} \in \mathbb{R}^{d_{\text{raw}} \times r}$ and $\boldsymbol{A} \in \mathbb{R}^{r \times k}$, with the rank $r \ll \min(d_{\text{raw}}, k)$. During fine-tuning, only the new low-rank matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ are updated while $\boldsymbol{W}$ remains frozen. The final adapted weight matrix $\boldsymbol{W}^*$ is then given by:

$$\boldsymbol{W}^* = \boldsymbol{W} + \Delta\boldsymbol{W} = \boldsymbol{W} + \boldsymbol{B}\boldsymbol{A} \tag{5}$$

This dramatically reduces the number of trainable parameters, enabling the RX to quickly adapt its model to the new data distribution using only the handful of training samples sent by the TX.

C. Overall Online Framework

The complete process, which integrates adaptive payload composition with rapid model fine-tuning, operates as an

Fig. 4. SIRIUS 5G-V2X testbed.

online framework illustrated in Fig. 3. The procedure is as follows:

1) *TX-side Processing:* The TX determines the transmission dimension α based on the channel state. It then compresses the test data $\boldsymbol{x}$ into a feature vector $\boldsymbol{z}$ using the PCA encoding process defined in (1). Concurrently, it calculates the maximum number of transmittable training samples, $N_{\max}$, using (2) and applies the same PCA compression to them.
2) *Transmission:* The TX transmits the compact, α-dimensional feature vectors $\boldsymbol{z}$ for both test and training samples, composed as described in Section III-A.
3) *RX-side Reconstruction:* Upon receiving the payload, the RX reconstructs the images $\hat{\boldsymbol{x}}$ for both datasets from their respective feature vectors using the inverse PCA transform shown in (3).
4) *LoRA Fine-Tuning:* Using the small set of reconstructed training data, the RX fine-tunes its classifier. This adaptation is performed efficiently using the LoRA mechanism detailed in Section III-B, resulting in an updated weight matrix $\boldsymbol{W}^*$ as shown in (5).
5) *Inference:* Finally, the RX feeds the reconstructed test data into the newly adapted model $\boldsymbol{W}^*$ to perform the final classification task.

IV. Experiments

A. Experimental Setup

Experiments were conducted using two SIRIUS 5G-V2X testbeds [6] as the TX and RX, as shown in Fig. 4. The system operated on a V2X sidelink channel with a 40 MHz bandwidth. The TX power was set to 23 dBm, and the distance between the TX and RX was 3 meters in a line-of-sight (LOS) environment. To simulate a dynamic channel where the data rate is limited, the modulation was changed from 256-QAM to 64-QAM. We used the MNIST dataset [7], for which the RX's *Multi-Layer Perceptron* (MLP) classifier [8] was pre-trained on data reconstructed from $d_{\text{pre}} = 200$ features. We evaluate our proposed method against End-to-End and Pre-Trained baselines.

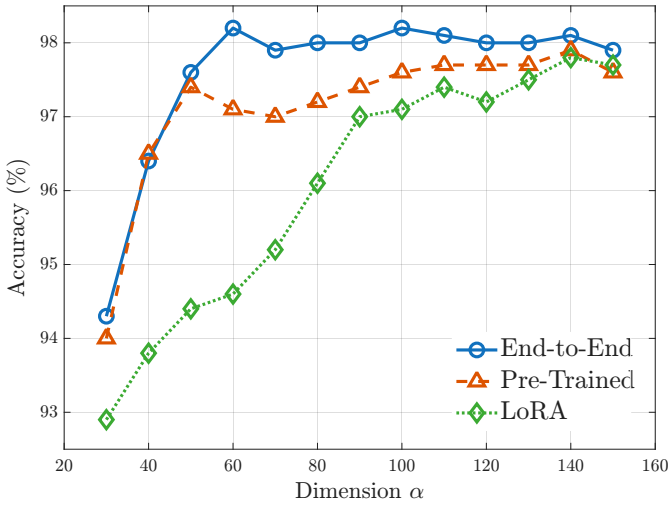

Fig. 5. Accuracy vs. Dimension for the MNIST dataset.

B. Results and Analysis

The performance of our proposed algorithm is compared against two baselines, as shown in Fig. 5 and Fig. 6.

First, we evaluate the classification accuracy, presented in Fig. 5. The three methods compared in the figure are as follows:

- **End-to-End:** For each transmission dimension α, a separate MLP model, including its entire weight matrix $\boldsymbol{W}$, is trained from scratch on data reconstructed specifically from α features. This represents the theoretical performance upper bound for that dimension but is impractical due to high latency and storage costs.
- **Pre-Trained:** The original pre-trained model $\boldsymbol{W}$, optimized for d_{pre}, is used directly to classify data reconstructed from the new dimension α without any adaptation. This baseline demonstrates the performance degradation caused by SMA.
- **LoRA (Proposed):** The original pre-trained model $\boldsymbol{W}$ is rapidly adapted using the LoRA fine-tuning technique described in Section III-B, utilizing the small set of training data to produce a final, adapted model $\boldsymbol{W}^*$.

The Pre-Trained baseline exhibits poor performance at lower dimensions and only gradually recovers as α increases, clearly illustrating the detrimental effect of SMA. In contrast, our proposed LoRA-based approach significantly outperforms this baseline across all tested dimensions. Notably, the accuracy of our method is comparable and, for a wide range of dimensions (e.g., 60 to 140), even slightly superior to the End-to-End baseline. This demonstrates the effectiveness of our rapid adaptation technique in recovering performance lost to SMA.

Next, we analyze the latency, presented in Fig. 6. The latency components shown in the figure are defined as follows:

- **Communication:** This measures the time from the TX initiating the payload calculation and PCA encoding to the RX completing the image reconstruction from the received feature vectors.

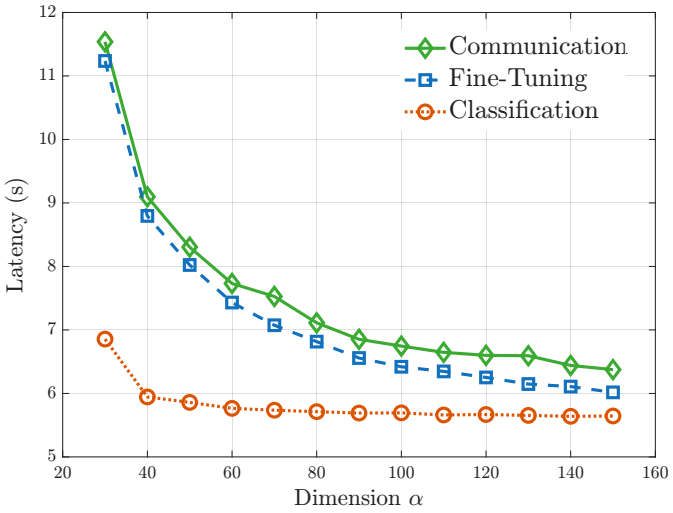

Fig. 6. Latency vs. Dimension for the MNIST dataset.

- **Fine-Tuning:** This is the cumulative time, including the Communication latency and the additional time required for the RX to adapt the pre-trained model using the LoRA technique.
- **Classification:** This represents the total end-to-end latency, encompassing all previous steps plus the final inference time for the classification task.

Fig. 6 shows a clear trade-off: as the transmitted dimension α decreases, more training data can be included in the payload, which slightly increases the fine-tuning time. Note that the latency for the End-to-End baseline is omitted from the figure, as the time required for full retraining is orders of magnitude higher than our fine-tuning approach, making a direct comparison on the same scale impractical and highlighting the significant time-saving advantage of our method.

V. Conclusion

This paper proposes a split-architecture SC algorithm that effectively addresses the SMA problem caused by varying channel conditions. By combining adaptive dimensionality reduction at the TX with rapid LoRA-based fine-tuning at the RX, our framework allows the system to adapt to channel changes in real-time. Experimental results on the MNIST dataset demonstrate that our approach achieves high task accuracy comparable to full retraining, but with significantly lower latency. This ensures both efficiency, by avoiding the computationally intensive process of retraining the entire model, and practicality, by enabling rapid adaptation to dynamic channel conditions.

Future work will proceed in two main directions. First, we will generalize the proposed framework. The MNIST dataset is relatively simple, allowing a basic MLP classifier to achieve high performance. We intend to validate our approach on more complex datasets that require advanced architectures, such as Convolutional Neural Networks (CNNs), to verify its effectiveness in more challenging scenarios. Second, we will develop a dynamic policy for optimal dimension control. Such a policy would aim to select the ideal transmission dimension α that maximizes the data rate for a given channel state, while ensuring that task performance remains above a predefined threshold.

Acknowledgment

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (No. RS-2024-00453301, RS-2025-02217000)

References

[1] M. Chen *et al.*, "Distributed learning in wireless networks: Recent progress and future challenges," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 12, pp. 3579–35 605, 2021.

[2] Z. Qin, X. Tao, J. Lu, W. Tong, and G. Y. Li, "Semantic communications: Principles and challenges," 2022. [Online]. Available: https://arxiv.org/abs/2201.01389

[3] J. Choi, J. Park, S.-W. Ko, J. Choi, M. Bennis, and S.-L. Kim, "Semantics alignment via split learning for resilient multi-user semantic communication," *IEEE Trans. Veh. Technol.*, vol. 73, no. 10, pp. 15 815–15 819, 2024.

[4] I. Jolliffe, *Principal Component Analysis*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 1094–1096. [Online]. Available: https://doi.org/10.1007/978-3-642-04898-2_455

[5] E. J. Hu *et al.*, "Lora: Low-rank adaptation of large language models." *Proc. Int. Conf. Learn. Representations*, vol. 1, no. 2, p. 3, 2022.

[6] Ettifos, "SIRIUS, 5G-V2X sidelink platform." [Online]. Available: https://www.ettifos.com/product-sirius

[7] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[8] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986.