Mixture-of-Experts for Multi-Task Semantic Communications with CSI-Free Multiple Access

Sujin Kook[1], Jihong Park[2], Seong-Lyun Kim[1], Seung-Woo Ko[3]
[1]Department of Electrical and Electronics Engineering, Yonsei University, Seoul, South Korea
[2]ISTD Pillar, Singapore University of Technology and Design, Singapore,
[3]Department of Smart Mobility Engineering, Inha University, Incheon, South Korea
[1]{sjkook, slkim}@ramo.yonsei.ac.kr, [2]jihong park@sutd.edu.sg, [3]swko@inha.ac.kr

Abstract

This work investigates the use of a *Mixture-of-Experts* (MoE) framework for *multi-task semantic communications* (MT-SemCom) in scenarios where multiple devices simultaneously transmit multi-task semantic features without channel state information (CSI). Two key design components are proposed. First, each device applies a random linear transformation to its data, which preserve the underlying semantic features while enabling reliable reconstruction provided that the resulting subspaces are approximately orthogonal. Second, to handle the case where subspaces partially overlap and inter-device interference arises, the receiver employs an MoE-based architecture with an additional multi-task expert trained to be robust against such interference. These complementary designs jointly deliver substantial gains for MT-SemCom, as validated through two-device simulations on a mixed MNIST and FMNIST dataset under CSI-free multiple access.

1 Introduction

Mixture-of-Expert (MoE) refers to an emerging deep learning architecture consisting of multiple neural network models called experts, each trained to handle specific tasks, and a gating network, trained to select the most suitable experts Jacobs et al. [1991]. An input task fed into the gating network is routed to a few experts deemed suited, whose outputs are combined to make the final result. This structure is effective in addressing complex problems broken down into multiple tasks, particularly in modern large foundation models to address a wide range of tasks Fan et al. [2022].

The effectiveness of MoE in managing multiple tasks well fits *multi-task semantic communication* (MT-SemCom), where only semantic features, extracted from input data samples, are transmitted to execute various subsequent tasks at a paired receiver Ma et al. [2018]. Several MoE-assisted designs on MT-SemCom have been introduced in the literature Chen et al. [2023a,b], Xue et al. [2024], summarized in the Appendix. Despite their effectiveness, these prior works overlook several challenges associated with supplying data to MoE through a wireless channel. Data samples, typically collected by many devices, should be forwarded to the MoE on time. To avoid excessive delay due to signaling overhead, *multiple access* (MA) without *channel state information* (CSI) is often favorable in practice, where multiple devices simultaneously access a wireless medium without channel-aware precoding/decoding Wen et al. [2023]. Interference due to collisions among them inevitably occurs, which may damage inherent data patterns required for the MoE's gating network to assign tasks to the tailored experts. Besides, the uplink channels vary over time, affecting the received signal's statistics as well as underlying data patterns. This coupling makes it increasingly challenging to identify and execute tasks accordingly.

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: AI and ML for Next-Generation Wireless Communications and Networking (AI4NextG).

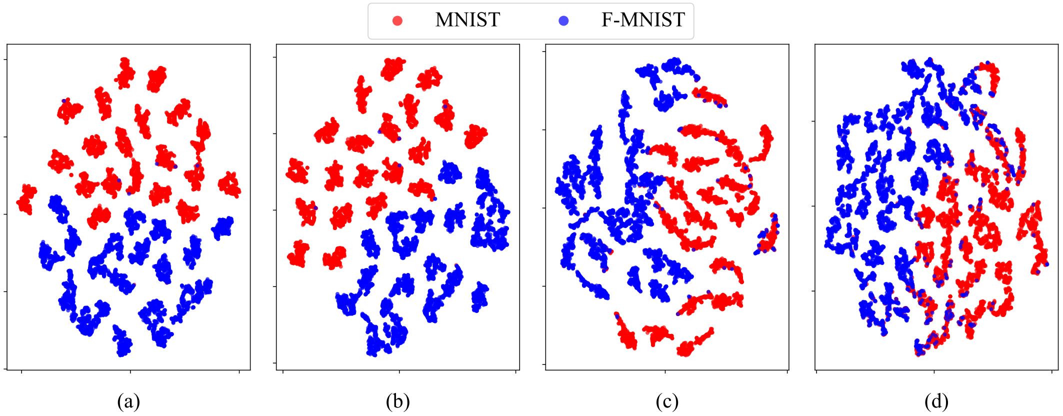

Figure 1: Graphical comparison between the original and the reconstructed SCs using 2D t-SNE Maaten and Hinton [2008] visualization for a two device MT-SemCom scenario with mixed MNIST and F-MNIST datasets. Fig. (a) shows the distribution of original SCs, while Figs. (b) to (d) illustrate the reconstructed SCs when the overlapping subspace dimension is 0, 1, and 2, respectively.

To address the aforementioned challenges, this work proposes a novel MT-SemCom framework, called *MoE for CSI-free MA* (MoEMA), which features two key designs as follows.

- **CSI-Free Semantic MA**: Each device transforms its multi-task semantic features into *semantic codewords* (SCs) by multiplying a random orthogonal matrix. After CSI-Free MA, through several post-processing techniques introduced in the sequel, each SC can be localized within a signal space spanned by one or a few devices' random matrices. Importantly, these subspaces are independent of channel realizations and can be detected from the received MA signals. Finally, we can reconstruct each SC by projecting the received signal onto the detected subspace, which reliably preserves the semantic feature as long as the involved subspaces are nearly orthogonal, as shown in the similarity between Fig. 1(a) and (b).
- **MoE-Based Interference Management**: The random matrix selection may lead to inter-device interference when the subspaces of different devices overlap (see Fig. 1(c) and (d)). Our MoE-based semantic receiver mitigates this effect by incorporating an additional multi-task expert that is specifically trained to manage signals corrupted by interference. The gating network then directs each SC to the multi-task expert when significant interference is detected, or to its corresponding exclusive expert otherwise.

The two designs—communication-for-AI and AI-for-communication—complement each other. The proposed CSI-free semantic MA preserves the inherent data patterns of the transmitted features, thereby facilitating the subsequent MoE operation. In turn, the MoE mitigates inter-device interference that a purely communication-oriented design cannot fully resolve in the absence of CSI. To the best of our knowledge, this is the first work to address MT-SemCom with CSI-free MA by jointly leveraging random precoding and MoE. The effectiveness of the proposed MoEMA has been validated by extensive simulations, showing an average test accuracy improvement of 4% over a benchmark single-expert system when the overlapping subspace dimension is two in an 3-dimensional signal space.

2 Mixture-of-Expert over CSI-Free Multiple Access: Design Overview

Consider a scenario of multi-device MT-SemCom, where devices in $\mathcal{N}$ have data for multiple tasks that will be delivered to the *access point* (AP) provisioned with a MoE server. Our goal is to develop an end-to-end design of MoE-based MT-SemCom under CSI-free MA, including data embedding and transformation at the device side, signal processing for data reconstruction at the AP side, and subsequent MoE for handling multiple tasks. Fig. 2 represents a graphical overview of the proposed design, outlined in the following.

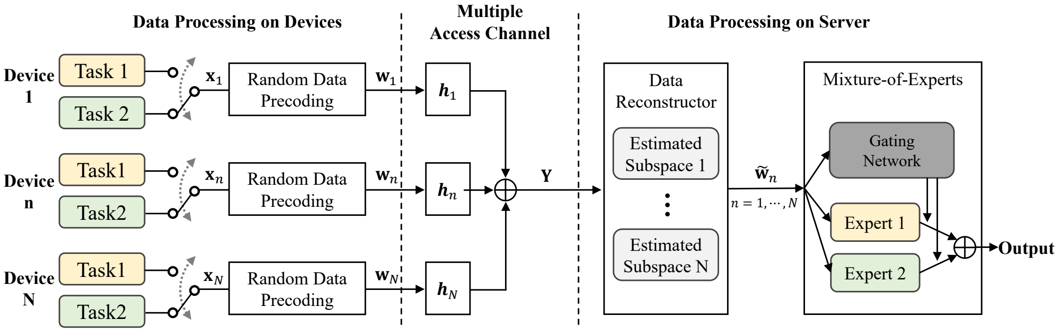

Figure 2: System overview of MoEMA with $|\mathcal{S}| = 2$. Time index t is omitted for simplicity.

2.1 Data Embedding, Device-Specific Random Data Transformation, and Auxiliary Signaling

First, we will explain the device-side preprocessing prior to transmission. A typical device, say device n, generates a raw data sample at time slot t, required to process one of the tasks in a list of $\mathcal{S}$. Once generated, it immediately passes through a pretrained embedding network and the resultant semantic feature is defined as $\mathbf{x}_n^{(t)} \in \mathbb{R}^{D\times 1}$, normalized to unit norm. We assume that the pretrained embedding network is given to all devices in $\mathcal{N}$, allowing all semantic features across devices to be expressible in a unified embedding space. Furthermore, to enhance the discriminability among task-specific embeddings, we design the embedding network using the ArcFace Deng et al. [2019], which explicitly enforces inter-task separation in the semantic space. The detailed configuration and training procedure are provided in Appendix B.

Next, device n transforms $\mathbf{x}_n^{(t)}$ to the corresponding SC, denoted by $\mathbf{w}_n^{(t)} \in \mathbb{R}^{M\times 1}$, by multiplying a device-specific random orthogonal matrix $\mathbf{A}_n \in \mathbb{R}^{M\times D}$, given as $\mathbf{w}_n^{(t)} = \mathbf{A}_n\mathbf{x}_n^{(t)}$ where $\mathbf{A}_n^\top \mathbf{A}_n = \mathbf{I}$. The SC $\mathbf{w}_n^{(t)}$ is a vector in the column space of $\mathbf{A}_n$, i.e., $\mathbf{w}_n^{(t)} \in \mathsf{span}(\mathbf{A}_n)$. We set $M \geq |\mathcal{N}|D$, and the subspace $\mathsf{span}(\mathbf{A}_n)$ has dimension D in an M-dimensional vector space. Each device's subspace, nearly orthogonal to the others due to the random matrix selection across devices, can serve as a unique signature to identify the corresponding device, as explained in the sequel.

In addition to transmitting the SCs, each device also sends an auxiliary signaling message to the server to further assist the gating network. This signaling is sent separately from the data transmission and is designed to convey task-related information extracted from the SCs. After training the embedding network with ArcFace, clusters corresponding to each task are formed, and the center of each cluster can be obtained; we refer to these as task-specific center points. Based on this, the auxiliary signals can represent either the distance between an SC and the task-specific center points or a one-hot indicator specifying the closest task-specific center. These auxiliary signals enhance the server's ability to perform accurate expert selection by providing explicit cues about the underlying task structure, which may not be fully preserved after the subspace transformation.

2.2 CSI-Free Multiple Access & Data Reconstruction

All devices' SCs at time t, say $\{\mathbf{w}_n^{(t)}\}_{n\in\mathcal{N}}$, are simultaneously transmitted to the AP over a shared wireless medium. We consider that each device has a single transmit antenna and the server is equipped with K receive antennas. The CSI from device n to the AP is thus modeled as a single-input multiple-output, denoted by $\mathbf{h}_n^{(t)} \in \mathbb{R}^{K\times 1}$. The received signal, denoted by $\mathbf{Y}^{(t)} \in \mathbb{R}^{K\times M}$, is

$$\mathbf{Y}^{(t)} = \sum_{n\in\mathcal{N}} \mathbf{h}_n^{(t)} \left(\mathbf{w}_n^{(t)}\right)^\top. \tag{1}$$

We assume a block fading model, where $\mathbf{h}_n^{(t)}$ remains constant within a time slot but varies independently across slots. All channel vectors are normalized to unit norm, i.e., $|\mathbf{h}_n^{(t)}| = 1$, and a thermal noise is ignored for simplicity.

Neither the devices in $\mathcal{N}$ nor the AP have knowledge of CSI $\{\mathbf{h}_n^{(t)}\}_n$. The received signal $\mathbf{Y}^{(t)}$ of (1) can thus be viewed as a superposition of multiple devices' SCs modulated by their random vectors. When $K > |\mathcal{N}|$ and the channels $\{\mathbf{h}_n^{(t)}\}$ are mutually independent, the transmitted SCs $\{\mathbf{w}_n^{(t)}\}_{n\in\mathcal{N}}$ remain embedded in the received vector's row space $\mathsf{row}(\mathbf{Y}^{(t)}) = \mathsf{span}\left((\mathbf{Y}^{(t)})^\top\right)$. This row space is contained within the aggregate signal space of the participating devices, $\mathsf{row}(\mathbf{Y}^{(t)}) \subseteq \mathsf{span}(\mathbf{A}_1, \cdots, \mathbf{A}_N)$. In other words, the collection of $\mathsf{row}(\mathbf{Y}^{(t)})$ across many slots approximates the full signal space, enabling reconstruction of all SCs by projecting $\mathbf{Y}^{(t)}$ onto the clustered subspaces, denoted as $\{\tilde{\mathbf{w}}_n^{(t)}\}_{n\in\mathcal{N}}$. The detailed step-by-step explanations will be provided in Sec. 3.

2.3 MoE for Multi-Task Processing with Reconstructed Data

The reconstructed SC $\tilde{\mathbf{w}}_n^{(t)}$ is sent to the edge server collocated with the AP and input into MoE therein, which has a tree structure comprising one gating network cascaded to $|\mathcal{S}|$ expert networks. When $\tilde{\mathbf{w}}_n^{(t)}$ reaches the MoE, the gating network first processes it to compute a logit vector $\mathbf{g}_n^{(t)} \in \mathbb{R}^{|\mathcal{S}|\times 1}$, which quantifies how well experts are suited to handle $\tilde{\mathbf{w}}_n^{(t)}$. Next, $\tilde{\mathbf{w}}_n^{(t)}$ is passed through all experts, and the final output is produced by combining their outputs using the logit vector $\mathbf{g}_n^{(t)}$. The detailed training and inference processes will be explained in Sec. 4.

3 CSI-Free Semantic Reconstruction

3.1 Signal Space Detection and Case Categorization

This subsection first explains our signal space detection methodology based on linear algebra. Then, we categorize two possible cases depending on the dimension of the detected signal space.

The received signal $\mathbf{Y}^{(t)}$ of (1), which is a sum of $|\mathcal{N}|$ rank-1 matrices, can be decomposed by applying a singular value decomposition, given as

$$\mathbf{Y}^{(t)} = \sum_{n=1}^{|\mathcal{N}|} \sigma_n \boldsymbol{\alpha}_n^{(t)} (\boldsymbol{\beta}_n^{(t)})^\top, \tag{2}$$

where σ_n is the n^{th} singular value, and $\boldsymbol{\alpha}_n^{(t)} \in \mathbb{R}^{K\times 1}$ and $\boldsymbol{\beta}_n^{(t)} \in \mathbb{R}^{M\times 1}$ are the corresponding right and left singular vectors, respectively. As aforementioned, all right singular vectors $\{\boldsymbol{\beta}_n^{(t)}\}_{n=1}^{|\mathcal{N}|}$ lie within the aggregate signal space $\mathsf{span}(\mathbf{A}_1, \cdots, \mathbf{A}_{|\mathcal{N}|})$, which is our target to detect in this subsection. To this end, we gather $\{\boldsymbol{\beta}_n^{(t)}\}_{n=1}^{|\mathcal{N}|}$ over T slots to make a matrix $\mathbf{B} \in \mathbb{R}^{M\times T|\mathcal{N}|}$. This matrix creates a subspace $\mathsf{span}(\mathbf{B})$, which asymptotically converges to $\mathsf{span}(\mathbf{A}_1, \cdots, \mathbf{A}_N)$ as T increases. For the detection of $\mathsf{span}(\mathbf{B})$, we compute an eigen decomposition of $\mathbf{B}\mathbf{B}^\top$, given as

$$\mathbf{B}\mathbf{B}^\top = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^\top, \tag{3}$$

where $\boldsymbol{\Lambda} \in \mathbb{R}^{M\times M}$ is a diagonal matrix of eigenvalues and $\mathbf{U} \in \mathbb{R}^{M\times M}$ includes the paired eigenvectors. Then, the orthonormal basis vectors of $\mathsf{span}(\mathbf{B})$ are the first r columns in $\mathbf{U}$, say $\mathbf{U}_{1:r}$, where r is the number of non-zero eigenvalues in $\boldsymbol{\Lambda}$, equivalent to the dimension of $\mathsf{span}(\mathbf{B})$.

Recall that the dimension of each device's subspace is D. Then, the dimension of $\mathsf{span}(\mathbf{B})$, say r, ranges from D to $|\mathcal{N}|D$ depending on how many subspaces overlap, as summarized below.

- **Case 1. No Overlap**: When $r = |\mathcal{N}|D$, each device's D-dimensional subspace can be represented separately in the aggregate signal space $\mathsf{span}(\mathbf{B})$. This can be easily detected using a clustering technique, which will be explained in Sec. 3.2.
- **Case 2. Overlap**: When $D \le r \le |\mathcal{N}|D$, overlapping and non-overlapping subspaces coexist in $\mathsf{span}(\mathbf{B})$. We address them separately, estimating the overlapping subspaces first, and then estimating the non-overlapping subspaces after excluding the previously identified overlapped ones. A detailed explanation will be given in Sec. 3.3.

3.2 Case 1: No Overlapping Subspaces between Devices

In this case, each column vector of $\mathbf{B}$ should belong to only one subspace among $\mathsf{span}(\mathbf{A}_1), \cdots, \mathsf{span}(\mathbf{A}_{|\mathcal{N}|})$. In other words, each device's subspace can be retrieved by clustering $\mathbf{B}$'s column vectors to form $|\mathcal{N}|$ numbers of low-dimensional subspaces. We use a *sparse* *subspace clustering* (SSC) technique to this end, which is effective in dividing the entire vector into the specified number of subsets Elhamifar and Vidal [2013]. Each subset after SSC contains vectors that can be expressed as a sparse linear combination of the others, thereby regarded as one device's subspace. The detailed clustering process is omitted due to the space limit.

The estimated subspaces are denoted by $\{\mathsf{span}(\tilde{\mathbf{A}}_n)\}_{n=1}^{|\mathcal{N}|}$, where the columns of $\tilde{\mathbf{A}}_n$ represents its orthonormal basis vectors.[1] Last, we can reconstruct the corresponding SCs, denoted by $\{\tilde{\mathbf{w}}_n^{(t)}\}$, by projecting $\{\mathbf{Y}^{(t)}\}$ onto $\tilde{\mathbf{A}}_n$.

3.3 Case 2: Subspace Overlap between Devices

In this case, unlike the previous one, some column vectors of $\mathbf{B}$ may belong to multiple subspaces among $\mathsf{span}(\mathbf{A}_1), \cdots, \mathsf{span}(\mathbf{A}_{|\mathcal{N}|})$, making it challenging to estimate each device's subspace directly from $\mathsf{span}(\mathbf{B})$. Therefore, we will separate $\mathsf{span}(\mathbf{B})$ into overlapping and non-overlapping subspaces and process them sequentially. In this work, we focus on the scenario with two devices ($|\mathcal{N}| = 2$), while extending it to a general one with multiple devices in our future work.

First, we aim to identify the bases of the overlapping subspace by revisiting $\mathbf{\Lambda}$ and $\mathbf{U}$ specified in (3). The eigenvalue in $\mathbf{\Lambda}$ represents the total squared projection of the matrix $\mathbf{B}$'s all columns onto the corresponding eigenvector in $\mathbf{U}$. In essence, the more frequently an eigenvector is contributed to express $\mathbf{B}$'s column vectors, the larger its eigenvalue is likely to be. Consequently, the overlapping subspaces shared by multiple devices tend to have larger eigenvalues than non-overlapping ones. Note that the overlapping subspace's dimension is $\gamma = |\mathcal{N}|D - r$. We then select the γ columns from $\mathbf{U}$ associated with the γ largest eigenvalues in $\mathbf{\Lambda}$, and use them as basis vectors of the overlapping subspaces, denoted by $\mathbf{V} \in \mathbb{R}^{M \times \gamma}$.

Next, we remove the component of $\{\mathbf{Y}^{(t)}\}$ that lies in the overlapping subspace by projecting each $\mathbf{Y}^{(t)}$ onto the null space of $\mathbf{V}$:

$$\hat{\mathbf{Y}}^{(t)} = \mathsf{proj}_{\mathsf{null}(\mathbf{V})}\mathbf{Y}^{(t)} = \mathbf{Y}^{(t)}(\mathbf{I} - \mathbf{V}\mathbf{V}^\top). \tag{4}$$

The resulting matrices $\{\hat{\mathbf{Y}}^{(t)}\}$ have $|\mathcal{N}|$ right singular vectors residing in the non-overlapping signal space $\mathsf{span}(\mathbf{A}_1, \cdots, \mathbf{A}_{|\mathcal{N}|}) \cap \mathsf{span}(\mathbf{V})^\perp$. Collecting these right singular vectors over T slots yields $\hat{\mathbf{B}} \in \mathbb{R}^{M \times T|\mathcal{N}|}$, whose column vectors span the remaining signal space $\mathsf{span}(\hat{\mathbf{B}})$. Similar to the preceding case, we use SSC to partition this space into $|\mathcal{N}|$ subspaces, denoted by $\{\mathsf{span}(\mathbf{C}_n)\}_{n=1}^{|\mathcal{N}|}$ where each $\mathbf{C}_n \in \mathbb{R}^{M \times (D-\gamma)}$ contains basis vectors of the device n's non-overlapping subspace. Last, stacking $\mathbf{V}$ and $\mathbf{C}_n$ becomes the basis vectors of the device n's total subspace, namely, $\tilde{\mathbf{A}}_n = [\mathbf{C}_n : \mathbf{V}]$, and the corresponding SC can be reconstructed following the approach in Sec. 3.2.

4 Training and Inference of MoE

MoE training begins after collecting the reconstructed SCs $\{\tilde{\mathbf{w}}_n^{(t)}\}$ over T slots, each associated with the predefined task in $\mathcal{S}$. To ensure efficient training, we adopt a two-phase approach. In the first phase, the experts are trained according to their designated roles: as described in Sec. 2.3, we employ $|\mathcal{S}|$ task-specific experts, training using the SCs corresponding to their tasks respectively. This stage promotes early specialization and establish a strong foundation. In the second phase, we fine-tune the entire MoE—both the gating network and all experts—using the complete set of reconstructed SCs $\{\tilde{\mathbf{w}}_n^{(t)}\}$. This joint optimization enables the gating network to effectively associate each SC with the most suitable expert, even when reconstructions are imperfect due to CSI-free MA.

The training objective of MoE is to minimize the combined losses of task-specific performance and load balance. The task-specific loss varies depending on the type of the concerning task, e.g., cross entropy for classification, mean squared loss for regression, and reconstruction loss for signal recovery. On the other hand, the load-balance loss helps regularize the expert selection behavior, encouraging the gating network to distribute the loads across multiple experts rather than collapsing to a single expert. This approach promotes both specialization and robustness. The load balance loss

[1] We assume that a one-to-one mapping between the detected subspaces and the devices in $\mathcal{N}$ is achievable by assigning a device-specific basis vector to one column of $\mathbf{A}_n$ in advance.

Table 1: Effect of overlapping dimension on subspace similarity and MoE performance.

Overlapping Dimension	(a) Cosine Similarity		(b) Learning Performance			
	Subspace	Data	w/o subspace estimation		w/ subspace estimation	
			Single	MoEMA	Single	MoEMA
0	0.99	0.99	93.23	93.26	93.18	**93.22**
1	0.99	0.95	88.35	89.66	88.00	**89.61**
2	0.99	0.89	72.0	76.15	71.98	**76.07**

is excluded in the first phase. However, it is included in the second phase with a weighting factor of λ. This hyperparameter controls the trade-off between prediction accuracy and expert diversity.

Upon completing the MoE training, MoEMA automatically processes the MA signal $\mathbf{Y}^{(t)}$ of (1) for $t > T$. This procedure involves reconstructing SCs from the estimated subspaces, routing them to appropriate experts via the gating network, and executing the corresponding task at the selected expert. This framework enables low-latency multi-task inference without requiring explicit coordination among devices or tasks.

5 Experiment Results

5.1 Experiment Setting

Dataset We use the well-known MNIST and Fashion-MNIST (F-MNIST) datasets to construct a multi-task scenario. MNIST classes are labeled from 0 to 9, while F-MNIST classes are re-indexed from 10 to 19. Each device is assigned data samples from both datasets in equal proportion to ensure a uniform task distribution across devices, with $15,000$ disjoint training samples and $5,000$ test samples per device.

Model The MoE model in our experiments comprises three expert networks: Expert 1 and Expert 2 are pretrained on MNIST and F-MNIST, respectively. Then, the gating network and all experts are trained together on the mixed dataset. Each expert network is implemented as a fully connected neural network with three hidden layers. The gating network consists of two hidden layers. All networks use ReLU activations and are optimized using the Adam optimizer.

5.2 Results

In MT-SemCom over a CSI-free MA environment, overlapping subspaces between devices act as a source of interference. As observed in Table 1 (a), the subspace similarity remains relatively high across different levels of overlap, suggesting that the interference caused by overlapping subspaces has limited impact on the reconstruction of the subspace itself. In contrast, data-level similarity degrades notably as the overlap increases, indicating that overlapping basis vectors hinder the separability of reconstructed SCs. The exact definitions of the similarity metrics are provided in the Appendix.

Despite imperfect reconstruction, the proposed MoEMA framework effectively mitigates interference through its MoE structure. We evaluate MoEMA against **Single**, a unified neural network trained on the combined dataset without any task- or device-specific separation. As shown in Table 1 (b), MoEMA outperforms the Single benchmark by about 4%, demonstrating its ability to maintain high performance even under significant overlap—unlike a unified model that processes all data at once. The performance obtained using different signaling strategies is provided in the Appendix.

Furthermore, the performance gap between the w/o subspace estimation and w/ subspace estimation settings is marginal across all overlap levels, as shown in Table 1 (b). In this comparison, w/o subspace estimation represents the ideal case where the receiver is assumed to know the true transmit-side subspaces, while w/ subspace estimation reflects the practical setting in which the receiver must reconstruct the subspaces from the aggregated signal. The fact that their performance has a small gap indicates that the clustering-based subspace estimation is highly accurate, even without prior knowledge of the true subspaces. In other words, the reconstructed subspaces obtained through unsupervised clustering are sufficiently precise to enable the MoE router to select the appropriate experts almost as reliably as in the ideal case where the true subspaces are known. This result underscores the practicality of MoEMA, demonstrating that it can operate effectively in realistic settings where the server does not have access to device-specific transformation matrices.

6 Conclusions

This study has investigated the problem of enabling reliable MT-SemCom in CSI-free multiple-access environments. A key requirement for achieving robust performance in such settings is i) to mitigate inter-device interference and ii) to preserve task-relevant structure throughout the transmission pipeline. The proposed MoEMA framework was designed to satisfy both requirements by combining device-specific subspace transformations with the MoE. The former assigns each device a distinct subspace signature, while the latter adaptively selects task-appropriate experts to counteract interference and representation mismatch.

Comprehensive analytic and numerical studies show that MoEMA maintains high performance even under significant subspace overlap, and that clustering-based reconstruction achieves accuracy comparable to the ideal case where the device subspaces are known at the server. These findings confirm the effectiveness and practicality of MoEMA as a scalable solution for multi-task semantic communication in realistic CSI-free environments.

MoEMA can be extended toward more diverse device configurations, richer task distributions, and dynamic network conditions. Another promising direction includes developing adaptive subspace learning, online subspace tracking, and end-to-end joint optimization of the embedding network and MoE router. These extensions will further enhance the scalability and generalizability of MoEMA in broader multi-device and multi-task scenarios.

Acknowledgments

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No.RS-2024-00404972, Development of 5G-A vRAN Research Platform) and the National Research Foundation of Korea (NRF) grant funded by the Korea government(MSIT) (RS-2025-02217000).

References

Tianlong Chen, Xuxi Chen, Xianzhi Du, Abdullah Rashwan, Fan Yang, Huizhong Chen, Zhangyang Wang, and Yeqing Li. Adamv-moe: Adaptive multi-task vision mixture-of-experts. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17346–17357, 2023a.

Zitian Chen, Yikang Shen, Mingyu Ding, Zhenfang Chen, Hengshuang Zhao, Erik G Learned-Miller, and Chuang Gan. Mod-squad: Designing mixtures of experts as modular multi-task learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11828–11837, 2023b.

Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4690–4699, 2019.

Ehsan Elhamifar and René Vidal. Sparse subspace clustering: Algorithm, theory, and applications. *IEEE transactions on pattern analysis and machine intelligence*, 35(11):2765–2781, 2013.

Zhiwen Fan, Rishov Sarkar, Ziyu Jiang, Tianlong Chen, Kai Zou, Yu Cheng, Cong Hao, Zhangyang Wang, et al. M^3vit: Mixture-of-experts vision transformer for efficient multi-task learning with model-accelerator co-design. *Advances in Neural Information Processing Systems*, 35:28441–28457, 2022.

Jiayi He, Xiaofeng Luo, Jiawen Kang, Hongyang Du, Zehui Xiong, Ci Chen, Dusit Niyato, and Xuemin Shen. Toward mixture-of-experts enabled trustworthy semantic communication for 6g networks. *IEEE Network*, 2024.

Sin-Yu Huang, Renjie Liao, and Vincent WS Wong. Leveraging moe-based large language model for zero-shot multi-task semantic communication. *arXiv preprint arXiv:2503.15722*, 2025.

Robert A Jacobs, Michael I Jordan, Steven J Nowlan, and Geoffrey E Hinton. Adaptive mixtures of local experts. *Neural computation*, 3(1):79–87, 1991.

Guangyuan Liu, Yinqiu Liu, Jiacheng Wang, Hongyang Du, Dusit Niyato, Jiawen Kang, Zehui Xiong, and Abbas Jamalipour. Context-aware semantic communication for the wireless networks. *arXiv preprint arXiv:2505.23249*, 2025.

Jiaqi Ma, Zhe Zhao, Xinyang Yi, Jilin Chen, Lichan Hong, and Ed H Chi. Modeling task relationships in multi-task learning with multi-gate mixture-of-experts. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 1930–1939, 2018.

Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

Hai Pham, Young Jin Kim, Subhabrata Mukherjee, David P Woodruff, Barnabas Poczos, and Hany Hassan Awadalla. Task-based moe for multitask multilingual machine translation. *arXiv preprint arXiv:2308.15772*, 2023.

Anke Tang, Li Shen, Yong Luo, Nan Yin, Lefei Zhang, and Dacheng Tao. Merging multi-task models via weight-ensembling mixture of experts. *arXiv preprint arXiv:2402.00433*, 2024.

Dingzhu Wen, Xiang Jiao, Peixi Liu, Guangxu Zhu, Yuanming Shi, and Kaibin Huang. Task-oriented over-the-air computation for multi-device edge ai. *IEEE Transactions on Wireless Communications*, 23(3):2039–2053, 2023.

Nan Xue, Yaping Sun, Zhiyong Chen, Meixia Tao, Xiaodong Xu, Liang Qian, Shuguang Cui, and Ping Zhang. Wdmoe: Wireless distributed large language models with mixture of experts. In *GLOBECOM 2024-2024 IEEE Global Communications Conference*, pages 2707–2712. IEEE, 2024.

A Related Works

MoE has recently gained attention as a promising approach for supporting MT-SemCom. As briefly mentioned in the introduction, MT-SemCom aims to extract and transmit only semantic features from data to support multiple downstream tasks at the receiver side. MoE is well-suited for such scenarios due to its inherent ability to handle task diversity and expert specialization. In this appendix, we summarize several recent studies that have leveraged MoE in MT-SemCom frameworks.

For example, zero-shot MT-SemCom has been proposed in Huang et al. [2025] by utilizing a pretrained LLM's MoE, which enables the system to handle a variety of tasks, including unseen ones. In He et al. [2024], MoE has been exploited to address heterogeneous adversarial attacks on SemCom that are infeasible with a single expert. In Liu et al. [2025], a pair of transmitter and receiver shares an identical MoE to synchronize data and expert selections depending on the given context awareness, such as service requirements and channel information.

MoE has recently emerged as a promising paradigm to strengthen MT-SemCom, as it can flexibly allocate model capacity across tasks. This capability is particularly important because most MT models are large and often suffer from task interference when learning MT simultaneously, which typically leads to performance degradation. In Chen et al. [2023a], MoE is employed to support visual recognition by enabling task-dependent backbones, thereby mitigating the instability of using a single unified backbone for multiple tasks. In Chen et al. [2023b], MoE is integrated into vision transformers to improve computational efficiency by activating only task-relevant experts during training, which effectively modularizes the network and facilitates MT scalability. Likewise, Tang et al. [2024] leverages MoE to compress large foundation models by selectively activating a subset of experts, thus making MT adaptation more efficient. These efforts highlight that MoE is inherently effective in capturing task relevance at the sample level and dynamically selecting suitable experts to enhance performance. Beyond vision tasks, Xue et al. [2024] demonstrates that decomposing *large language models* (LLMs) into a gating network at the base station and distributed expert modules across devices can further extend capabilities to resource-constrained environments. Finally, Pham et al. [2023] introduces task-specific adaptors to efficiently add new tasks into MoE. however, this approach primarily focuses on task separation and does not explicitly consider the feature-level characteristics within each task.

B Embedding Network with ArcFace

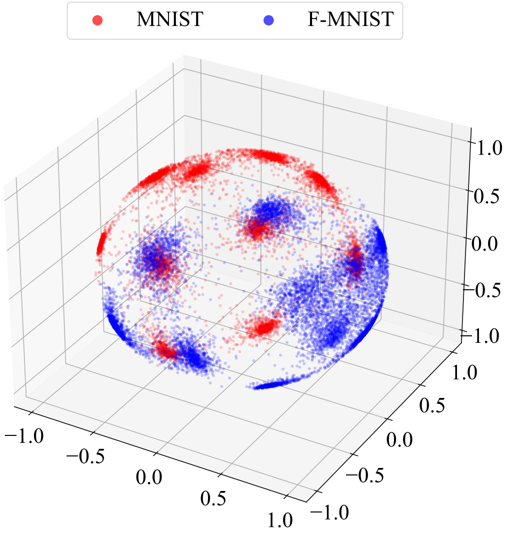

Figure 3: The visualization of embedding results.

To construct a unified semantic embedding space while ensuring separability across different tasks, we employ an embedding network trained with the ArcFace loss Deng et al. [2019]. ArcFace introduces an angular margin between classes, enabling the embedding network to generate representations that

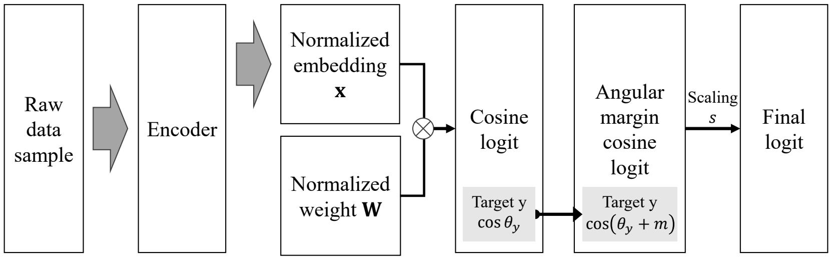

Figure 4: The computation pipeline of the embedding network with ArcFace head.

are not only compact within each task cluster but also well separated across tasks. This property is particularly beneficial for MoEMA, as the gating network relies on such geometric separation to perform effective expert routing.

Fig. 4 illustrates the computation pipeline for training the embedding network. After passing through the encoder, each sample is mapped to a latent embedding vector $\mathbf{x}$, which is ℓ_2-normalized to lie on a unit hypersphere. The ArcFace head also normalizes its weight vectors $\mathbf{W}$, making the angular relationship between $\mathbf{x}$ and $\mathbf{W}_y$ the key factor in classification. Under this formulation, the classifier computes the cosine logit

$$\cos\theta_y = \mathbf{x}^\top \mathbf{W}_y. \tag{5}$$

To enhance angular discrimination, the standard cosine logit is replaced with an additive angular-margin formulation. Specifically, the target-class logit is modified as $\alpha \cdot \cos(\theta_y + m)$, where α is a scaling factor and m is the angular margin. Introducing this margin effectively reduces the classifier's confidence during training, encouraging the encoder to learn embeddings with greater angular separability. As a result, ArcFace promotes larger inter-task angular margins, yielding more discriminative features that improve MoEMA's expert routing performance. After training is completed, the ArcFace head is discarded, and only the encoder is used to generate semantic embeddings for transmission.

In our implementation, the encoder is an MLP with two hidden layers and the embedding dimension is set to $D = 3$. We use an angular margin of $m = 0.5$ and a scaling factor of $s = 16$. As observed in our experiments, the use of ArcFace significantly enhances the task-discriminative properties of the transmitted SCs, thereby improving the robustness of MoEMA in CSI-free MA environments. The embedding result is shown in Fig. 3.

C Experiment Details

This section provides a comprehensive overview of the experimental setup used in our study. All experiments were conducted on a system equipped with an Intel Core i7-9700 CPU and an NVIDIA GeForce RTX 2080 Ti GPU. Model development was carried out using PyTorch version 3.11.3 Paszke et al. [2019]. The detailed hyperparameter settings are summarized in Table 2. Unless otherwise specified, the following system parameters are used throughout all experiments. Each device embeds data into an 3-dimensional space, which is then transformed into a 10-dimensional representation. The server is equipped with 16 antennas.

D Similarity Metrics

This appendix defines the two complementary metrics used in Table 1 (a) to evaluate reconstruction performance: subspace similarity and data similarity. The subspace similarity quantifies the geometric alignment between the original subspace $\mathbf{A}_n$ and its estimate $\tilde{\mathbf{A}}_n$, while the data similarity measures the directional consistency between the transformed data $\mathbf{w}_n^{(t)}$ and its reconstruction $\tilde{\mathbf{w}}_n^{(t)}$. These metrics provide a comprehensive assessment of the reconstruction performance in both structural and sample-wise aspects.

Table 2: System parameters

Parameter	Value
Embedding dimension (D)	3
Projected dimension (M)	10
Receiver antennas (K)	16
Batch size	32
Learning rate	0.001
Training epochs	200
Dropout rate	0.1

Subspace Similarity Given the orthonormal basis matrices $\mathbf{A}_n$ and $\tilde{\mathbf{A}}_n$ for the original and estimated subspaces, we define the corresponding projection matrices as $\mathbf{P}_n = \mathbf{A}_n \mathbf{A}_n^\top$ and $\tilde{\mathbf{P}}_n = \tilde{\mathbf{A}}_n \tilde{\mathbf{A}}_n^\top$. The subspace similarity is then computed using the Frobenius norm as:

$$\text{Subspace Similarity}(\mathbf{A}_n, \tilde{\mathbf{A}}_n) = \frac{\|\mathbf{P}_n \tilde{\mathbf{P}}_n\|_F^2}{D}, \tag{6}$$

where $\|\cdot\|_F$ denotes the Frobenius norm and D is the subspace dimension. This formulation ensures that the similarity score lies between 0 and 1, with higher values indicating greater subspace alignment.

Data Similarity To measure the similarity between the original transformed feature $\mathbf{w}_n^{(t)}$ and its reconstructed counterpart $\tilde{\mathbf{w}}_n^{(t)}$, we adopt the cosine similarity metric, defined as:

$$\text{Sim}(\mathbf{w}_n^{(t)}, \tilde{\mathbf{w}}_n^{(t)}) = \frac{\mathbf{w}_n^{(t)} \cdot \tilde{\mathbf{w}}_n^{(t)}}{\|\mathbf{w}_n^{(t)}\| \, \|\tilde{\mathbf{w}}_n^{(t)}\|}. \tag{7}$$

This metric captures the angular similarity between the two vectors, with a value close to 1 indicating strong directional alignment.

E Auxiliary Signaling Strategies

Table 3: Effect of signaling strategies on MoEMA performance. Two signaling strategies are considered: (i) distance-based, which sends the distance from an SC to the task-specific center points, and (ii) one-hot, which converts this distance information into a one-hot indicator of the nearest center.

Overlapping Dimension	w/o subspace estimation		w/ subspace estimation	
	distance-based	one-hot	distance-based	one-hot
0	93.26	**93.29**	93.22	**93.23**
1	89.66	**89.71**	**89.61**	89.46
2	76.15	**76.19**	**76.07**	75.76

To assist the gating network, each device transmits an auxiliary signaling feature along with the SC. In this work, we evaluate two different strategies for constructing this signaling feature:

- **Distance-based signaling**: the Euclidean distance between the transmitted SC and a task-specific center point.
- **One-hot signaling**: a one-hot indicator representing the closest task-specific center.

Both strategies aim to provide lightweight task-discriminative cues that may not be fully preserved after the subspace transformation. Despite this conceptual difference, Table 3 show that these two forms of signaling yield nearly identical performance within the MoEMA framework. This is because the MoE router primarily exploits the coarse task-level structure rather than the exact numerical form of the auxiliary feature. As long as the signaling conveys which task a given SC is most closely aligned with, both approaches enable the gating network to perform accurate expert selection.

Therefore, the choice between distance-based and one-hot signaling can be made based on implementation convenience or system constraints, without significant impact on MoEMA's overall performance.