FedRNL: Federated Rationalization with Soft Parameter Sharing

Lingxiao Kong, Jiahui Jiang, Haozhao Wang, Lei Wu (✉), and Ruixuan Li

School of Computer Science and Technology, Huazhong University of Science and Technology Wuhan, China
leiwu@hust.edu.cn

Abstract. Interpretability is crucial in natural language processing to enhance transparency and trust. Rationalization models achieve this by extracting key input fragments, i.e., rationales, to explain decisions while preserving predictive performance. On the other side, Federated Learning (FL) is recently emerging as a key paradigm for training machine learning models because it can leverage training data from multiple clients without the requirement of uploading their original data. Considering this, we firstly propose training Rationalization models in a FL manner. However, we find that simply combining them suffers from serious performance degradation due to the data heterogeneity among clients, where there exists inconsistent rationale generation. To solve this issue, we propose FedRNL which introduces a soft-sharing mechanism to align generator and predictor encoders, ensuring shallow-consistency and deep-generalization. An encoder loss minimizes feature discrepancies, and a layer-wise aggregation strategy separately updates the generator and predictor at the server, enhancing model stability. Extensive experiments show that FedRNL significantly improves the performance as compared to existing general heterogeneity mitigation methods.

Keywords: Federated learning · Rationalization · Non-IID.

1 Introduction

Interpretability is crucial in natural language processing (NLP) as it enhances model transparency, fosters user trust, and supports decision-making [19, 31, 26]. However, deep learning models are often black-box systems, making their reasoning processes challenging to understand, thus necessitating better interpretability in NLP models. Rationalization models address this need by extracting key input fragments (i.e., rationales) that explain model decisions while preserving predictive performance [15, 5, 24]. They consist of a generator and a predictor: the generator selects the most informative text subset as rationales, which is then passed to the predictor for final classification or decision-making. However, existing research focuses on centralized implementations of these models, leaving their adaptation to distributed learning scenarios largely unexplored. As concerns about data privacy and security continue to rise, extending rationalization models to distributed environments such as Federated Learning (FL) has

become a crucial direction. However, the challenge posed by non-independent and identically distributed (non-IID) data complicates this transition.

In FL, data heterogeneity arises as different clients collect data from distinct sources, leading to significant distribution shifts. This non-IID nature results in inconsistent learning directions across clients, degrading the overall model performance and causing potential model drift. This issue is particularly pronounced for Rationalization models, as variations in textual styles, feature distributions, or annotation standards across clients. This inconsistency makes it challenging for the model to generate stable and coherent rationales, weakening its interpretability and generalization ability.

Many approaches have been proposed to address the non-IID issue, including parameter regularization [17, 2], personalized FL [33, 40], and local model alignment [10, 29]. While these approaches alleviate non-IID issues to some extent, optimizing federated learning performance for rationalization models while preserving their interpretability remains an open problem. Luo [21] found that the classifier has the lowest feature similarity between local models and suggested that bias in FL can be mitigated solely by rectifying the deep-networks (the classifier) of the deep network after federated training. On the other hand, Liu [20] found that maintaining shallow-networks consistency between the generator and predictor enables the model to learn more informative rationales. This contradiction raises an important question: *is deep-networks classifier calibration more critical, or does shallow-networks generator-predictor consistency play a bigger role in addressing non-IID challenges? More importantly, can we reduce classifier bias while preserving interpretability to further enhance Rationalization models in federated learning?*

To answer this question, we propose FedRNL, the first work to study training rationalization models through FL. Our goal is to mitigate the accuracy performance degradation of rationalization models in FL which is caused by non-IID. Specifically, FedRNL shares the encoder layers between the generator and predictor in a soft manner, and introduces an encoder loss to minimize the parameter inconsistency between encoders while allowing the deep network to adapt to the non-IID data distribution. Each client optimizes its local model independently while following the soft-sharing constraint to align generator and predictor representations. At the server, we adopt a layer-wise aggregation strategy, separately updating the generator and predictor parameters to enhance the stability of the global model further. The main contributions of this paper are:

- To the best of our knowledge, this is the first work to study training the rationalization models in a FL manner.
- We propose a soft-sharing mechanism between the generator and predictor. This mechanism ensures an adaptable consistency in the shallow network and allows the deep network to adapt to the non-IID data distribution, enhancing the model's generalizability and interpretability in FL.
- We provide a theoretical analysis that guarantees the convergence of the proposed method. Besides, the experimental results demonstrate that existing FL methods are unsuitable for rationalization tasks, while FedRNL signifi-

cantly improves accuracy while preserving the interpretability under various non-IID settings.

2 Related Work

Rationalization Models. Rationalization models aim to enhance interpretability in NLP by selecting key input fragments (rationales) that justify model predictions while maintaining predictive performance [15, 24]. These models typically consist of a generator that extracts the most informative text subset and a predictor that makes the final decision based on the selected rationales. As research on rationalization models advances, they can be further categorized into abstract and extractive approaches. Extractive rationalization models identify and extract keywords or sentences from the input text, capturing the most salient features to explain predictions [5, 6]. Some works study extractive methods using an encoder-decoder framework [15, 3]. The encoder assigns each word in the input sequence a binary tag to indicate whether it is part of the rationale. The decoder then processes only the highlighted rationale words and maps them to target categories [35]. Other works use attention mechanisms to extract rationales [34, 39]. Abstractive rationalization models generate rationales by constructing explanations using new words and restructured sentences from the input text [30]. Some works study text-to-text methods, which utilize sequence-to-sequence translation models, incorporating both the label and explanation simultaneously [27, 12]. Others use generative methods, which generate a free-form explanation and then make a prediction based on the produced abstractive rationale [4]. Despite significant progress, most rationalization models are designed for centralized training, assuming access to a single dataset. The challenge of adapting them to FL remains largely unexplored.

Non-IID in Federated learning. Many existing studies have proposed solutions to mitigate client drift caused by non-IID data [29, 25, 36]. Some approaches rely on personalization techniques to tailor model optimization to each client's unique data distribution [28, 40]. For instance, FedROD [7] integrates a globally shared general model with personalized client models through a joint training mechanism. FedProto [33] leverages prototypes—central representations of classes—to facilitate personalized model training for each client. There are also some works on improving the generalization ability of the global model, such as parameter regularization [17, 1] and local model alignment methods [41, 28]. For example, FedProx [18] introduces a proximal term to the local subproblem, adjusting local updates to account for the discrepancy between the global and local models. FedPer [2] combines each client with globally shared model parameters. FedNH [10] mitigates the impact of data heterogeneity on federated learning by incorporating class prototypes into the global model. However, these methods are not applicable to Rationalization models.

3 Preliminary

Rationalization. We use rationalization models, denoted as $R(x)$, which consists of two components: the generator and the predictor. The generator generates an extractive rationale $z = gen(x)$ for an input paragraph x. Then, the generated rationale z is employed in the class prediction of x. The prediction process can be formulated as $\hat{y} = pre(x, z)$. The whole process, i.e., class prediction with the rationale, can be integrated as one model and denoted as $R(x) = pre(gen(x)))$. FR [20] divides the structure of $gen(\cdot)$ and $pre(\cdot)$ into encoder layers $encoder_g(\cdot)$ and linear layers $linear_g(\cdot)$. The generator and predictor share the same $encoder_g(\cdot)$, ensuring shallow network consistency.

Federated Learning. We consider there are N clients, and each client i ($i \in [1, N]$) has a local dataset $D_i = \{(x_{ij}, y_{ij})\}_{j=1}^{m_i}$ where m_i is the data number of dataset D_i. The non-IID of FL represents that the distribution of D_i differs among clients. We denote $G(\cdot; \delta)$ represent global model and $L_i(\cdot; \theta, \phi)$ represent local model of client i. The local loss function is denoted by $\mathcal{L}$, which is used to optimize the local model on the local dataset D_i. After each communication round t ($t \in [1, T]$), the server aggregates the updates from each client by weighted averaging to update the global model. The standard FL process typically follows the FedAvg [23]. Each client i trains its local model on its own dataset D_i and computes the local model update, the local update process of local model is:

$$\theta_i^{t+1} = \theta_i^t - \eta \nabla_{\theta_i} \mathcal{L}(\theta_i^t, D_i), \quad \phi_i^{t+1} = \phi_i^t - \eta \nabla_{\phi_i} \mathcal{L}(\phi_i^t, D_i) \tag{1}$$

where η is the learning rate. The server aggregates the local model updates from all clients with weighted averaging, where m_i is the dataset size of i:

$$\delta^{t+1} = \sum_{i=1}^{m} \frac{m_i}{m} \delta_i^t \tag{2}$$

where $m = \sum_{i=1}^{N} m_i$ is the total number of samples across all clients.

4 Methodology

In this section, we propose a simple yet effective method named FedRNL to mitigate the model drift. Specifically, the core idea is to adopt a soft-sharing mechanism between the generator and predictor, ensuring consistency in the shallow network while allowing the deep network to adapt to the non-IID data distribution (i.e. shallow-consistency and deep-generalization). This approach enables the Rationalization model to maintain its interpretability while improving classification performance in non-IID scenarios, thereby achieving better generalization ability. An overview of the proposed framework is shown in Figure 1. The algorithm workflow is presented in Algorithm 1.

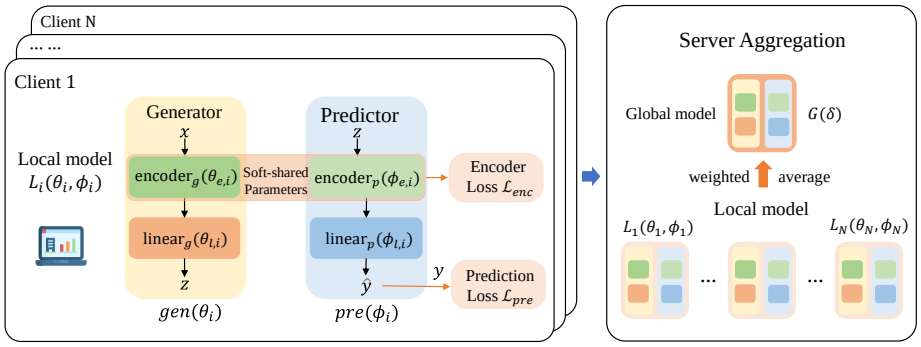
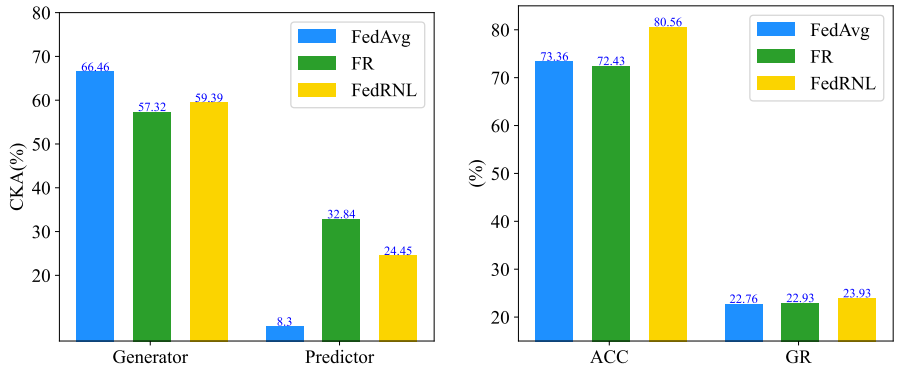


Fig. 1: The overall architecture of `FedRNL`.

4.1 Motivation

In FL, non-IID increases the difficulty of model training, particularly in Rationalization models that consist of a Generator and a Predictor. Existing FR methods[20] aim to enhance the interpretability of rationalization models by enabling the Generator and Predictor to share the same Encoder, thereby improving their consistency in shallow networks. However, the effectiveness of this approach in a FL setting remains unclear. To evaluate FR's performance in FL, we use the Gold Rationale F1 (GR) to measure the model's interpretability, Test Accuracy(ACC) to measure the model's prediction performance, and Centered Kernel Alignment (CKA) [14] similarity to assess the representation similarity between the Generator and Predictor across different clients' local models. As shown in Figure 2, FR successfully reduces the similarity between generators while increasing the similarity between predictors, improving consistency in shallow networks. However, it does not effectively mitigate the non-IID issue—ACC even drops by 0.93% compared to FedAvg, and the improvement in interpretability is minimal. The core idea of the FR is to enhance model interpretability through shallow network consistency. **However, experimental results indicate that forcibly sharing the shallow network alone does not significantly improve interpretability. Moreover, it may restrict the expressiveness of the deep network, making it difficult for the model to adapt to the non-IID data distribution in the FL.**

Thus, we propose a key question: can we design a method that ensures consistency in the shallow network to maintain the interpretability of rationalization models while allowing the deep network to adapt flexibly to non-IID data distributions? In FL, the non-IID problem requires models to achieve strong generalization across different data distributions. If the generator and predictor share an identical shallow network, they may lack the necessary flexibility to adapt to diverse local data distributions, ultimately affecting overall performance. Therefore, a new strategy is needed—one that ensures shallow network consistency to enhance interpretability, while enabling the deep network to achieve greater generalization, allowing it to adapt effectively to non-IID data distributions.



(a) CKA similarity (b) Model performance

Fig. 2: Impact of non-IID over different methods.

4.2 Local Model Update

In the FedRNL framework, each client i trains its local model L_i using the dataset D_i it has access to, while maintaining privacy and avoiding direct sharing of raw data. For simplicity, we will omit the i marker in the subsequent sections of this section. The local model L consists of two components: the generator $gen(\cdot;\theta)$ and the predictor $pre(\cdot;\phi)$. We further decompose these two components into encoder layers and linear layers. Specifically, the generator $gen(\cdot;\theta_e,\theta_l)$ includes the $encoder_g(\cdot;\theta_e)$ and $linear_g(\cdot;\theta_l)$, while the predictor $pre(\cdot;\phi_e,\phi_l)$ consists of the $encoder_p(\cdot;\phi_e)$ and $linear_p(\cdot;\phi_l)$.

The generator is responsible for producing a rationale $z = gen(x;\theta_e,\theta_l)$ from the input data x, where z represents a subset or transformation of the input that is deemed most relevant for the prediction. First, the generator's encoder layer extracts the semantic features from the input text x and generates a feature representation $r_g = encoder_g(x;\theta_e)$ for each token of x. r_g are then passed to the generator's linear layer and generate a mask $z = linear_g(r_g;\theta_l)$ to select the most important parts of the text for the classification task. Through this process, the generator creates the rationale z, providing the necessary information for subsequent prediction. Then z is passed into the encoder layer of the predictor, which is responsible for extracting the feature representation $r_p = encoder_p(z;\phi_e)$ of z. r_p are then processed through the linear layer of the predictor, followed by the classification layer, which performs the classification task and outputs the final prediction $\hat{y} = linear_p(r_p;\phi_l)$. This process enables the predictor to make accurate predictions based on the rationale provided by the generator.

The whole process, i.e., class prediction with the rationale, can be integrated as one model and defined as $L(x) = pre(gen(x;\theta_e,\theta_l),\phi_e,\phi_l)$. To train the performance of the rationalization model, we calculate the cross-entropy loss between the predictions $\hat{y}$ outputted by the predictor and the ground-truth labels y. This loss is referred to as prediction loss $\mathcal{L}_{pre}$. The optimization constraint used in

the training process is defined as follows:

$$\mathcal{L}_{pre} = \sum_{(x,y)\in\mathbb{D}_i} \mathcal{L}(L(x;\theta_e,\theta_l,\phi_e,\phi_l),y) \tag{3}$$

where the $\mathcal{L}$ represents the cross-entropy loss of the rationalization model.
Encoder loss. To enforce feature alignment between the generator and predictor, we introduce a soft-sharing mechanism that aligns the encoder layer learned by the generator and the predictor across clients. Specifically, we design an encoder loss $\mathcal{L}_{enc}$ to control the parameter similarity between the encoder layers $encoder_g$ and $encoder_p$, aimed at optimizing the parameters of these two encoders, ensuring that their feature representations remain consistent. The optimization constraint is defined as follows:

$$\mathcal{L}_{enc} = \|\theta_e - \phi_e\|_2 \tag{4}$$

where $\|\cdot\|_2$ denotes the squared L2 norm, which calculates the Euclidean distance between the two parameters. By minimizing this loss, the parameters of $encoder_g$ and $encoder_p$ are guided to optimize towards more similar directions, improving the coordination between the generator and the predictor and, ultimately, enhancing the performance of the model facing non-iid problem.
Joint loss. The total loss for each client is the sum of the prediction loss and the encoder loss:

$$\mathcal{L} = \mathcal{L}_{pre} + \lambda\mathcal{L}_{enc} \tag{5}$$

where λ is a hyperparameter that controls the balance between the prediction loss and the encoder loss.

4.3 Global Model Aggregation

In aggregating the global model, we use the FedAvg algorithm to aggregate the local model parameters from all clients. At round t, the server receives local model parameters from each client and updates the global model parameters. Specially, the parameters of the generator and predictor are aggregated separately to update the global model. The global aggregation is formulated as:

$$\sigma_g^{(t+1)} = \sum_{i=1}^{N} \frac{m_i}{m}\theta_i^{(t)}, \quad \sigma_p^{(t+1)} = \sum_{i=1}^{N} \frac{m_i}{m}\phi_i^{(t)} \tag{6}$$

where $\sigma_g^{(t+1)}$ and $\sigma_p^{(t+1)}$ represent the global model parameters of the generator and predictor. $\theta_i^{(t)}$ represent the local model parameters of generator, which consist of encoder layer parameters $\theta_{e,i}^{(t)}$ and linear layer parameters $\theta_{l,i}^{(t)}$. $\phi_i^{(t)}$ represent the local model parameters of predictor, which consist of encoder layer parameters $\phi_{e,i}^{(t)}$ and linear layer parameters $\phi_{l,i}^{(t)}$.

Algorithm 1 Training Process of FedRNL

Input: clients N, local dataset $\mathbb{D}_i, i = 1, \ldots, N$, communication rounds T, local epoch M, learning rate η, encoder weight λ

Global server does:

Initialize global model $G(.;\delta)$
for $t = 1$ to T **do**
  Select N_t participated clients
  for each client $i \in [N_t]$ **in parallel do**
    $L_i(.;\theta_i,\phi_i) \leftarrow$ LocalUpdate$(i, G(\delta))$
  end for
  Aggregate global model $G(.;\delta)$ by Eq.(6)
end for

LocalUpdate$(i, G(\delta))$:

for $m = 1$ to M **do**
  for batch $(x_{ij}, y_{ij} \in \mathbb{D}_i)$ **do**
    Compute loss using λ by Eq. (5)
    Update local model $L_i(.;\theta_i,\phi_i)$ according to the loss
  end for
end for
return $L_i(.;\theta_i,\phi_i)$

5 Theoretical Analysis

We make the following assumptions for these objectives, which are widely adopted in FL [32, 36].

Assumption 1 ***(L-smoothness).*** *The objective function $\mathcal{L}_i$ is L-smooth with Lipschitz constant $L > 0$, i.e., $\|\nabla\mathcal{L}_i(\theta,\phi) - \nabla\mathcal{L}_i(\theta',\phi')\|_2 \leq \mathcal{L}_i(\|\theta-\theta'\| + \|\phi-\phi'\|)_2$ for all $\theta, \phi, \theta', \phi'$.*

Assumption 2 ***(Bounded Variance).*** *For all parameters θ, ϕ, the variance of the local stochastic gradient in each client is bounded by σ^2, i.e., $\mathbb{E}\|\nabla\mathcal{L}_i(\theta,\varphi) - \nabla\mathcal{L}(\theta,\varphi)\|^2 \leq \sigma^2$.*

Assumption 3 ***(Bounded diversity).*** *Under non-IID data distribution, the variance of local gradients to global gradient is bounded by ζ^2, i.e., $\|\nabla\mathcal{L}_i(\theta_i,\varphi_i) - \nabla\mathcal{L}_g(\sigma_g,\sigma_p)\|^2 \leq \zeta^2$.*

Based on Assumption 1, we have

$$\begin{aligned}\mathcal{L}_g(\sigma_g^{t+1},\sigma_p^{t+1}) \leq \mathcal{L}_g(\sigma_g^t,\sigma_p^t) + \langle\nabla\mathcal{L}_g(\sigma_g^t,\sigma_p^t),(\sigma_g^{t+1}-\sigma_g^t,\sigma_p^{t+1}-\sigma_p^t)\rangle \\ + \frac{L}{2}\|(\sigma_g^{t+1}-\sigma_g^t,\sigma_p^{t+1}-\sigma_p^t)\|^2\end{aligned} \tag{7}$$

Based on Assumption 2 and 3, we have

$$\mathbb{E}\mathcal{L}_g(\sigma_g^{t+1},\sigma_p^{t+1}) \leq \mathbb{E}\mathcal{L}_g(\sigma_g^t,\sigma_p^t) - \eta\|\nabla\mathcal{L}_g(\sigma_g^t,\sigma_p^t)\|^2 + \frac{L\eta^2}{2}(\sigma^2+\zeta^2). \tag{8}$$

Based on the above, we have the following theory for the convergence of the proposed algorithm.

Theorem 1. *If the learning rate η diminishes with $O\left(\frac{1}{\sqrt{T}}\right)$, then the global model achieves asymptotic convergence, i.e.,*

$$\mathbb{E}\mathcal{L}_g(\sigma_g^T, \sigma_p^T) - \mathcal{L}_g^* \leq \frac{1}{\sqrt{T}}\left(\mathcal{L}_g(\sigma_g^0, \sigma_p^0) - \mathcal{L}_g^*\right) + \frac{L}{2\sqrt{T}}(\sigma^2 + \zeta^2). \tag{9}$$

The details of the proof can be found in Appendix A.

6 Experiments

Datasets. Following [20], we implemented the performance on two widely used rationalization dataset: **Beer Reviews** [22] and **Hotel Reviews** [37], which is a token-level multi-aspect sentiment analysis dataset for beer. Our experiments employ four aspects: Appearance, Palate, Taste, and Aroma. The training set, development set, and test set consist of 33782, 8731, and 936 available examples. The federated learning system comprises five clients for non-IID settings, each exclusively owning data sampled from one of five distinct datasets.
Evaluation Metrics. For task performance evaluation, we employ *classification accuracy (ACC)* and *Gold Rationale F1 (GR)*. ACC is used to assess classification performance by comparing the predicted class label with the actual label. GRchen2022can, defined as the F1 score between the predicted and human-annotated rationale, is used to evaluate the quality of rationale generation. A higher GR score signifies a more substantial alignment between the model-generated and Gold rationale, indicating better interpretability of the model.
Configurations. We employ the GRU-base models [9] to encode text, which has been adopted by most previous works [11, 38, 20]. We use Adam optimizerD-BLP:journals/corr/KingmaB14 for model training. The max sequence length, the network dropout rate, the sparsity trade-off, and the continuity trade-off are set to 256, 0.2, 10, and 10, respectively. The learning rate η, the communication round T, the batch size, and the hidden dims are set to 1×10^{-5}, 500, 512, and 200, respectively. Unless otherwise mentioned, each client's local epoch M is set to 5, and encoder weight λ is set to 2. Our models are trained with NVIDIA GeForce RTX 3090 (Ubuntu 22.04 LTS PyTorch).
Baselines. We compare FedRNL with several popular and state-of-the-art FL methods: FedAvg [23], FedProx [18], FedBABU [28], FedDyn [1], and Moon [16].

6.1 Performance Evaluation

To evaluate the effectiveness of FedRNL, we conduct experiments on the Beer dataset, which concludes 4 aspects: Appearance, Palate, Taste, and Aroma, and the Hotel dataset, which concludes 3 aspects: Cleanliness, Location, and Service. We assess model performance on the validation sets of these aspects independently. Additionally, to measure the model's generalization ability across different aspects, we construct a mixed validation set by randomly sampling 50% of

Table 1: Best test accuracy(%) and GR(%) over Beer datasets. **Bold** fonts highlight the best accuracy.

Methods	Appearance		Palate		Taste		Aroma		Average	
	GR	ACC	GR	ACC	GR	ACC	GR	ACC	GR	ACC
FedAvg	22.77	72.73	22.76	73.92	22.76	73.02	22.76	73.75	22.76	73.36
FedProx	22.43	72.51	22.45	72.51	22.44	73.21	22.44	72.96	22.44	72.80
FedBABU	22.93	66.54	22.94	68.01	22.94	67.33	22.94	67.25	22.94	67.28
FedDyn	22.28	75.8	22.25	75.89	22.28	75.97	22.32	76.2	22.28	75.97
Moon	22.83	77.57	22.84	77.52	22.84	77.6	22.84	77.55	22.84	77.56
FedRNL	**23.92**	**80.56**	**23.93**	**80.67**	**23.95**	**80.42**	**23.94**	**80.59**	**23.93**	**80.56**

Table 2: Best test accuracy(%) and GR(%) over Hotel datasets. **Bold** fonts highlight the best accuracy.

Methods	Cleanliness		Location		Service		Average	
	GR	ACC	GR	ACC	GR	ACC	GR	ACC
FedAvg	14.72	87.97	14.70	86.95	14.75	87.29	14.72	87.40
FedProx	14.98	87.63	15.01	87.80	14.97	86.95	14.99	87.46
FedBABU	15.09	87.12	15.11	87.12	15.06	87.97	15.09	87.40
FedDyn	15.02	83.05	14.97	82.54	14.98	82.71	14.99	82.77
Moon	14.35	85.08	14.36	84.75	14.38	84.92	14.36	84.92
FedRNL	**15.52**	**89.66**	**15.78**	**90.68**	**15.50**	**89.15**	**15.60**	**89.83**

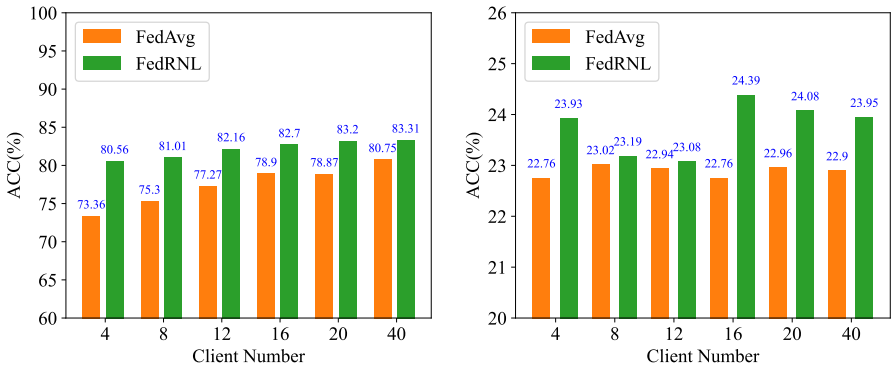
the data from each validation set and reporting the average performance. Table 1 and Table 2 demonstrate the superior performance of FedRNL compared to traditional federated learning methods across the Beer and Hotel datasets. In Table 1 (Beer dataset), FedRNL achieves the highest ACC of 80.56% and the best GR of 23.93% at the highest non-IID level, significantly outperforming other methods such as second-best method Moon (77.56% ACC, 22.84% GR) and FedAvg (73.36% ACC, 22.76% GR). Similarly, in Table 2 (Hotel dataset), FedRNL again achieves the best performance with an ACC of 89.83% and GR of 15.60% at the highest non-IID level, surpassing the second-best method, FedProx (87.46% ACC, 14.99% GR). These results indicate that FedRNL not only improves model accuracy but also enhances generalization across different datasets, effectively mitigating the significant challenges posed by non-IID data in federated learning and improving the model's interpretability.

Furthermore, the results on the mixed validation set confirm that FedRNL maintains strong generalization capabilities when handling data from multiple aspects. Unlike FedBABU, which exhibits weak performance in specific categories such as Appearance and Taste (66.54% and 67.33%, respectively), FedRNL achieves a more balanced and consistent improvement across different aspects. This suggests that the soft-sharing strategy between the generator and predictor ensures consistency in the shallow network while allowing the deep

network to adapt to the non-IID data distribution. The substantial performance gap between FedRNL and other methods further highlights its robustness in heterogeneous federated learning scenarios.

6.2 Ablation Study

To further verify the contributions of our proposed method, we conduct ablation studies in different settings. The learning rate of the model training and other settings remains consistent.

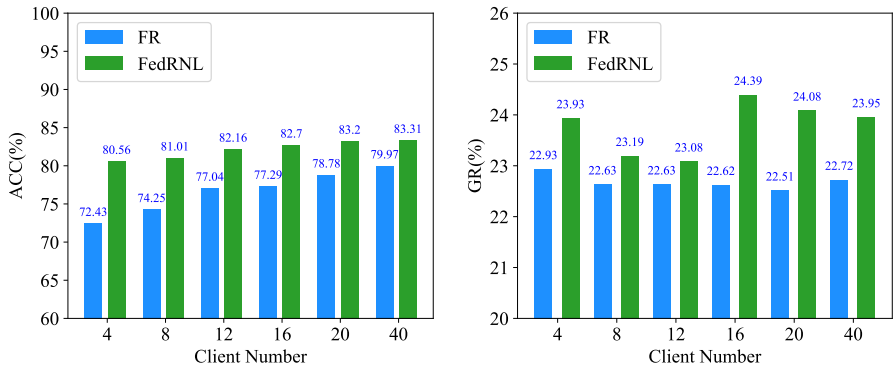


(a) ACC (b) GR

Fig. 3: Model performance comparison under different non-IID levels.

Impact of different non-IID level. To evaluate the performance of FedRNL under different non-IID levels (i.e., different numbers of clients), we compared the ACC and GR of FedRNL with FedAvg across varying numbers of clients on Beer dataset. We partitioned each aspect data into 10 subsets, resulting in a total of 40 subsets while ensuring that each subset maintains the same label distribution. Each client randomly selects one subset for training. Clients originating from the same dataset are categorized as IID clients, whereas those from different datasets are considered non-IID clients. As the number of clients decreases, the degree of non-IID increases, leading to more pronounced data heterogeneity. Figure 3a presents the ACC across different client settings, demonstrating that FedRNL consistently outperforms FedAvg, with the performance gap becoming more pronounced in highly non-IID scenarios. Specifically, when the number of clients is 4 (highest non-IID level), FedRNL achieves 80.56% accuracy, significantly surpassing FedAvg (73.36%), indicating its superior ability to mitigate performance degradation caused by data heterogeneity. As the number of clients increases, the non-IID effect weakens, and the accuracy of both methods improves, but FedRNL maintains a consistent advantage. Figure 3b shows that FedRNL consistently achieves higher GR scores than FedAvg, with a peak GR of 23.93% at 4 clients, compared to 22.76% for FedAvg. This suggests that FedRNL not only

improves accuracy but also enhances interpretability by generating rationales that better align with human annotations.



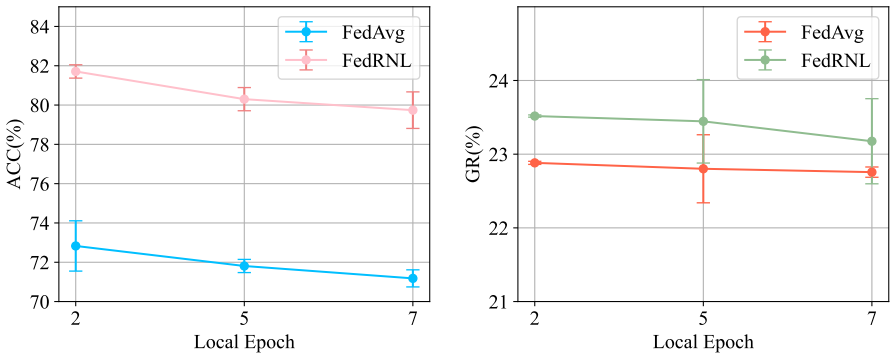
(a) ACC (b) GR

Fig. 4: Model performance comparison under different sharing ways.

Impact of different sharing ways. To analyze FedRNL's effectiveness compared to FR, we evaluate performance under different non-IID levels on the Beer dataset. FR uses a hard-sharing experimental setup that uses an identical encoder, meaning that the encoder parameters of both the generator and the predictor are completely the same. Figure 4a shows that FedRNL consistently outperforms FR in ACC, demonstrating the advantages of soft-sharing encoder parameters instead of enforcing full parameter sharing. When the number of clients is 4 (highest non-IID level), FedRNL achieves 80.56% accuracy, significantly outperforming FR's 72.43%, indicating that FR sharing limits the model's adaptability to non-IID data, whereas FedRNL's soft-sharing approach enables better feature extraction and generalization across different client distributions. Figure 4b indicates that FedRNL consistently achieves higher GR scores than FR across all client settings. The gap is most prominent at 4 clients (highest non-IID level), where FedRNL attains 23.93% compared to FR's 22.93%, demonstrating that soft-sharing encoders improve rationale alignment in highly non-IID level.

Impact of different local epoch. We evaluate the impact of varying the number of local epochs on ACC and GR on the Beer dataset. Figure 5 shows that FedRNL consistently outperforms FedAvg, demonstrating superior model performance. However, as the number of local epochs increases from 2 to 7, both FedRNL and FedAvg exhibit a slight decline in ACC and GR, which may be attributed to local overfitting caused by excessive local updates in a highly non-IID setting. Despite this decline, FedRNL maintains a significant accuracy and interpretability advantage over FedAvg across all settings, highlighting its robustness in FL scenarios.

Impact of different encoder loss. To evaluate the effect of different encoder loss functions on model performance, we compare FedAvg under different



(a) ACC (b) GR

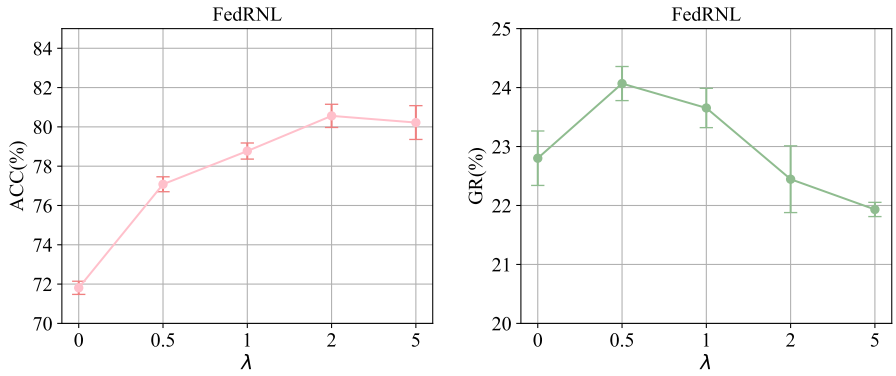
Fig. 5: Model performance comparison under different local epoch.

Table 3: Best test accuracy(%) and GR(%) using different encoder loss over Beer dataset. **Bold** fonts highlight the best accuracy.

Loss	4		8		12		16		20		40	
	GR	ACC	GR	ACC	GR	ACC	GR	ACC	GR	ACC	GR	ACC
FedAvg	22.76	73.36	23.02	75.30	22.94	77.27	22.76	78.90	22.46	78.87	22.90	80.75
Cos	22.43	75.13	22.68	76.65	22.71	77.88	22.80	78.81	22.59	80.28	22.86	81.15
L1	20.27	78.76	22.52	79.74	21.13	64.86	24.30	80.30	20.03	70.26	**24.54**	80.92
L2	21.23	76.14	22.46	77.32	21.67	80.25	21.78	81.37	21.80	82.72	21.62	82.95
Frobenius	**23.93**	**80.56**	**23.19**	**81.01**	**23.08**	**82.16**	**24.39**	**82.70**	**24.08**	**83.20**	23.95	**83.31**

non-IID levels with four loss functions: Cosine Similarity (Cos), L1 loss, L2 loss, and Frobenius norm loss. Table 3 shows that across different non-IID levels, the Frobenius norm loss consistently achieves the highest accuracy, outperforming other loss functions, including FedAvg. For instance, with 40 clients, the Frobenius loss reaches 83.31%, which is the best among all configurations. And the L1 loss also performs relatively well. The GR values show a similar trend, where Frobenius loss achieves the highest GR across different non-IID levels, only 0.59% below the L1 function at 40 clients. Experimental results demonstrate the ability of our proposed encoder loss to guarantee model interpretability and alleviate non-IID problems. And Frobenius loss is the most effective encoder loss function, as it provides the best balance between accuracy and generalization.

Impact of different encoder weight. To evaluate the impact of the encoder weight on FedRNL's performance, we conduct experiments with varying values of λ over Beer dataset. Figure 6a shows that ACC improves consistently as λ increases, reaching a peak at approximately 80%. This suggests that incorporating encoder alignment enhances model consistency and improves performance. However, further increasing λ to 5 results in a slight decline in ACC, indicating that over-constrained representations may restrict the model's learning flexibility and hinder adaptation to diverse client distributions. Figure 6b reveals a



(a) ACC (b) GR

Fig. 6: Model performance comparison under different encoder weight.

non-monotonic trend in GR. The generalization ratio initially increases, peaking at $\lambda = 0.5$ with a value above 24%, demonstrating that moderate encoder alignment improves the quality of rationales. However, GR starts to decline as λ further increases beyond 1, suggesting that excessive alignment may lead to over-constrained representations, reducing rationale diversity and quality.

7 Conclusion

In this work, we introduce FedRNL, the first method to adapt rationalization models to federated learning (FL) while addressing the challenges of non-IID data. By incorporating soft-sharing between generator and predictor encoders, we ensure consistency in the shallow network while allowing the deep network to adapt to the non-IID data distribution. Additionally, the encoder loss function ensures feature alignment, while our layer-wise aggregation strategy improves robustness in global model updates. Our theoretical results guarantee the convergence of `FedRNL`. Experimental results on benchmark datasets demonstrate that FedRNL significantly improves both classification accuracy and rationale quality, effectively bridging the gap between interpretability and FL robustness.

Acknowledgments. This work is supported by the National Key Research and Development Program of China under grant 2024YFC3307900; the National Natural Science Foundation of China under grants 62376103, 62302184, 62436003 and 62206102; Major Science and Technology Project of Hubei Province under grant 2024BAA008; Hubei Science and Technology Talent Service Project under grant 2024DJC078; and Ant Group through CCF-Ant Research Fund.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Acar, D.A.E., Zhao, Y., Navarro, R.M., Mattina, M., Whatmough, P.N., Saligrama, V.: Federated learning based on dynamic regularization. In: 9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021. OpenReview.net (2021)
2. Arivazhagan, M.G., Aggarwal, V., Singh, A.K., Choudhary, S.: Federated learning with personalization layers. arXiv preprint arXiv:1912.00818 (2019)
3. Arous, I., Dolamic, L., Yang, J., Bhardwaj, A., Cuccu, G., Cudré-Mauroux, P.: Marta: Leveraging human rationales for explainable text classification. In: AAAI. vol. 35, pp. 5868–5876 (2021)
4. Atanasova, P.: Generating fact checking explanations. In: Accountable and Explainable Methods for Complex Reasoning over Text, pp. 83–103. Springer (2024)
5. Bastings, J., Aziz, W., Titov, I.: Interpretable neural predictions with differentiable binary variables. In: Proceedings of the 57th AMACL. pp. 2963–2977. ACL Anthology (2019)
6. Chan, A., Sanjabi, M., Mathias, L., Tan, L., Nie, S., Peng, X., Ren, X., Firooz, H.: Unirex: A unified learning framework for language model rationale extraction. In: International Conference on Machine Learning. pp. 2867–2889. PMLR (2022)
7. Chen, H., Chao, W.: On bridging generic and personalized federated learning for image classification. In: The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022. OpenReview.net (2022)
8. Chen, H., He, J., Narasimhan, K., Chen, D.: Can rationalization improve robustness? In: 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2022. pp. 3792–3805. Association for Computational Linguistics (ACL) (2022)
9. Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using rnn encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078 (2014)
10. Dai, Y., Chen, Z., Li, J., Heinecke, S., Sun, L., Xu, R.: Tackling data heterogeneity in federated learning with class prototypes. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 37, pp. 7314–7322 (2023)
11. Huang, Y., Chen, Y., Du, Y., Yang, Z.: Distribution matching for rationalization. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 35, pp. 13090–13097 (2021)
12. Jang, M., Lukasiewicz, T.: Are training resources insufficient? predict first then explain! arXiv preprint arXiv:2110.02056 (2021)
13. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: Bengio, Y., LeCun, Y. (eds.) 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings (2015)
14. Kornblith, S., Norouzi, M., Lee, H., Hinton, G.: Similarity of neural network representations revisited. In: International conference on machine learning. pp. 3519–3529. PMLR (2019)
15. Lei, T., Barzilay, R., Jaakkola, T.: Rationalizing neural predictions. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. pp. 107–117 (2016)
16. Li, Q., He, B., Song, D.: Model-contrastive federated learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10713–10722 (2021)

17. Li, T., Hu, S., Beirami, A., Smith, V.: Ditto: Fair and robust federated learning through personalization. In: International conference on machine learning. pp. 6357–6368. PMLR (2021)
18. Li, T., Sahu, A.K., Zaheer, M., Sanjabi, M., Talwalkar, A., Smith, V.: Federated optimization in heterogeneous networks. Proceedings of Machine learning and systems **2**, 429–450 (2020)
19. Lipton, Z.C.: The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery. Queue **16**(3), 31–57 (2018)
20. Liu, W., Wang, H., Wang, J., Li, R., Yue, C., Zhang, Y.: Fr: Folded rationalization with a unified encoder. Advances in Neural Information Processing Systems **35**, 6954–6966 (2022)
21. Luo, M., Chen, F., Hu, D., Zhang, Y., Liang, J., Feng, J.: No fear of heterogeneity: Classifier calibration for federated learning with non-iid data. Advances in Neural Information Processing Systems **34**, 5972–5984 (2021)
22. McAuley, J.J., Leskovec, J., Jurafsky, D.: Learning attitudes and attributes from multi-aspect reviews. In: ICDM 2012. pp. 1020–1025. IEEE Computer Society (2012)
23. McMahan, B., Moore, E., Ramage, D., Hampson, S., y Arcas, B.A.: Communication-efficient learning of deep networks from decentralized data. In: Artificial intelligence and statistics. pp. 1273–1282. PMLR (2017)
24. Mendez Guzman, E., Schlegel, V., Batista-Navarro, R.: From outputs to insights: a survey of rationalization approaches for explainable text classification. Frontiers in Artificial Intelligence **7**, 1363531 (2024)
25. Meng, L., Qi, Z., Wu, L., Du, X., Li, Z., Cui, L., Meng, X.: Improving global generalization and local personalization for federated learning. IEEE Transactions on Neural Networks and Learning Systems (2024)
26. Mosbach, M., Gautam, V., Vergara-Browne, T., Klakow, D., Geva, M.: From insights to actions: The impact of interpretability and analysis research on nlp. arXiv preprint arXiv:2406.12618 (2024)
27. Narang, S., Raffel, C., Lee, K., Roberts, A., Fiedel, N., Malkan, K.: Wt5?! training text-to-text models to explain their predictions. arXiv preprint arXiv:2004.14546 (2020)
28. Oh, J., Kim, S., Yun, S.Y.: Fedbabu: Towards enhanced representation for federated image classification. arXiv preprint arXiv:2106.06042 (2021)
29. Qi, Z., Meng, L., Chen, Z., Hu, H., Lin, H., Meng, X.: Cross-silo prototypical calibration for federated learning with non-iid data. In: Proceedings of the 31st ACM International Conference on Multimedia. pp. 3099–3107 (2023)
30. Rajani, N.F., McCann, B., Xiong, C., Socher, R.: Explain yourself! leveraging language models for commonsense reasoning. In: ACL 2019 Volume 1: Long Papers. pp. 4932–4942 (2019)
31. Sun, X., Yang, D., Li, X., Zhang, T., Meng, Y., Qiu, H., Wang, G., Hovy, E., Li, J.: Interpreting deep learning models in natural language processing: A review. arXiv preprint arXiv:2110.10470 (2021)
32. T Dinh, C., Tran, N., Nguyen, J.: Personalized federated learning with moreau envelopes. Advances in neural information processing systems **33**, 21394–21405 (2020)
33. Tan, Y., Long, G., Liu, L., Zhou, T., Lu, Q., Jiang, J., Zhang, C.: Fedproto: Federated prototype learning across heterogeneous clients. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 36, pp. 8432–8440 (2022)
34. Vashishth, S., Upadhyay, S., Tomar, G.S., Faruqui, M.: Attention interpretability across nlp tasks. arXiv preprint arXiv:1909.11218 (2019)

35. Wang, H., Dou, Y.: Recent development on extractive rationale for model interpretability: A survey. In: 2022 International Conference on Cloud Computing, Big Data and Internet of Things (3CBIT). pp. 354–358. IEEE (2022)
36. Wang, H., Zheng, P., Han, X., Xu, W., Li, R., Zhang, T.: Fednlr: Federated learning with neuron-wise learning rates. In: Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. pp. 3069–3080 (2024)
37. Wang, H., Lu, Y., Zhai, C.: Latent aspect rating analysis on review text data: a rating regression approach. In: Rao, B., Krishnapuram, B., Tomkins, A., Yang, Q. (eds.) Proceedings of the 16th ACM SIGKDD, 2010. pp. 783–792. ACM (2010)
38. Yu, M., Zhang, Y., Chang, S., Jaakkola, T.: Understanding interlocking dynamics of cooperative rationalization. Advances in Neural Information Processing Systems **34**, 12822–12835 (2021)
39. Zhang, D., Sen, C., Thadajarassiri, J., Hartvigsen, T., Kong, X., Rundensteiner, E.: Human-like explanation for text classification with limited attention supervision. In: 2021 ieee international conference on big data. pp. 957–967. IEEE (2021)
40. Zhang, L., Fu, L., Liu, C., Yang, Z., Yang, J., Zheng, Z., Chen, C.: Towards few-label vertical federated learning. ACM TKDD (2024)
41. Zhu, Z., Hong, J., Zhou, J.: Data-free knowledge distillation for heterogeneous federated learning. In: International conference on machine learning. pp. 12878–12889. PMLR (2021)