FedVLN: Privacy-preserving Federated Vision-and-Language Navigation

Kaiwen Zhou and Xin Eric Wang

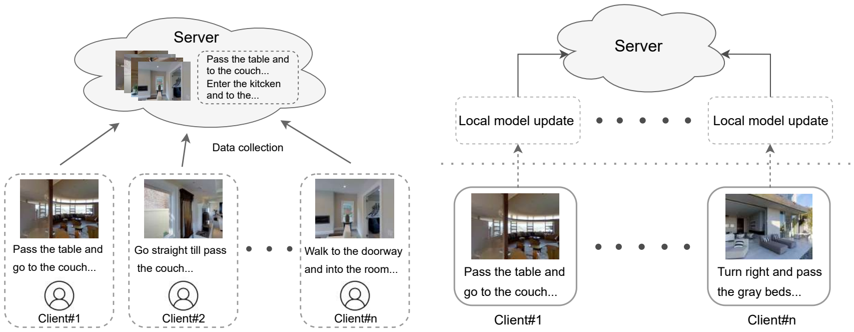
University of California, Santa Cruz, CA 95064, USA
kzhou35@ucsc.edu,xwang366@ucsc.edu

Abstract. Data privacy is a central problem for embodied agents that can perceive the environment, communicate with humans, and act in the real world. While helping humans complete tasks, the agent may observe and process sensitive information of users, such as house environments, human activities, etc. In this work, we introduce privacy-preserving embodied agent learning for the task of Vision-and-Language Navigation (VLN), where an embodied agent navigates house environments by following natural language instructions. We view each house environment as a local client, which shares nothing other than local updates with the cloud server and other clients, and propose a novel Federated Vision-and-Language Navigation (FedVLN) framework to protect data privacy during both training and pre-exploration. Particularly, we propose a decentralized federated training strategy to limit the data of each client to its local model training and a federated pre-exploration method to do partial model aggregation to improve model generalizability to unseen environments. Extensive results on R2R and RxR datasets show that, decentralized federated training achieve comparable results with centralized training while protecting seen environment privacy, and federated pre-exploration significantly outperforms centralized pre-exploration while preserving unseen environment privacy. Code is available at https://github.com/eric-ai-lab/FedVLN.

Keywords: Privacy-preserving Embodied AI, Vision-and-Language Navigation, Federated Learning

1 Introduction

A long-term goal of AI research is to build embodied agents that can perceive the environment, communicate with humans, and perform real-world tasks to benefit human society. However, since the agent interacts closely with humans and environments, it often receives sensitive information during training and inference. For example, as shown in Fig. 1(a), in the task of Vision-and-Language Navigation (VLN) [4], where an agent learns to navigate towards a target location in an indoor environment given natural language instruction, the training and inference data may include private information such as what the user's house looks like, what the user said, and what the user did. Data privacy is a central problem for building trustworthy embodied agents but seldomly studied


Server
Pass the table and to the couch... Enter the kitcken and to the...
Data collection
Pass the table and go to the couch...
Client#1
Go straight till pass the couch...
Client#2
Walk to the doorway and into the room...
Client#n
Server
Local model update
Local model update
Pass the table and go to the couch...
Client#1
Turn right and pass the gray beds...
Client#n

(a) Centralized VLN learning (b) Federated VLN learning

Fig. 1. Data privacy: centralized VLN learning *vs.* our federated VLN learning. Existing VLN approaches centralize all the client data in a server, including house environments and user instructions, which ignores users' privacy concerns. Our federated VLN framework keeps client data used only locally and the server receives nothing other than local model updates, so the client data privacy is preserved.

before [14]. Thus, in this work, we introduce privacy-preserving embodied agent learning for the task of vision-and-language navigation.

VLN models are typically trained on seen environments with ground-truth instruction-trajectory pairs and then deployed to unseen environments without any labeled data. After deployment, the agent may explore the unseen environment and adapt to the new environment for better performance, which is known as pre-exploration. However, as shown in Fig. 1(a), most of the existing methods assemble all the data in a server to train a navigation agent for both seen environment training and unseen environment pre-exploration. This is not practical in reality since users may not want to share the data in their own house due to privacy concerns. Privacy-preserving VLN requires the agent to protect the data privacy during both seen environment training and unseen environment pre-exploration while maintaining comparable navigation performance.

In this paper, we propose a novel Federated Vision-and-Language Navigation (FedVLN) framework, to address the aforementioned data privacy issues and improve the adaptation performance on unseen environments at the same time. Specifically, on the seen environment training stage, as shown in Fig. 1(b), we treat each house environment as a client. The client's local models (a VLN agent for navigation and a speaker for data augmentation) are trained on private local data, and then the model updates are sent to the server for model aggregation. No private data except the local model updates will be shared with the server and there is no communication between clients. During pre-exploration, we train the client models on seen environments and unseen environments simultaneously under federated learning paradigm—the client models do partial model aggregation (language encoder only) and partial local adaptation, enabling better adaptation to the local visual environment while maintaining general language understanding. Under our FedVLN framework, users do not need to share their data with any other party, thus the privacy of training data and inference data is protected.

Our experiments on Room-to-Room (R2R) [4] and Room-across-Room (RxR) [28][1] datasets validate the effectiveness of our framework. Our federated learning framework achieves comparable results with centralized training while preserving data privacy. More importantly, on the pre-exploration stage, we show that centralized pre-exploration hinders the agent from adapting to each specific environment, and our federated pre-exploration method achieves the best performance among prior pre-exploration methods such as centralized [42,40] and environment-based pre-exploration [11]. Our contributions are three-fold:

- We are the first to discuss data privacy concerns for vision-and-language navigation and define the privacy-preserving embodied AI problem for the two learning stages in VLN.
- We propose a novel federated learning framework for privacy-preserving VLN to ensure that users do not need to share their data to any party.
- Extensive results on R2R and RxR show that our federated learning framework not only achieves comparable results with centralized training, but also outperforms centralized and environment-based pre-exploration methods.

2 Related Work

2.1 Vision-and-Language Navigation

With the development of deep learning and human's vision of more helpful AI agents, embodied AI becomes an emerging research area. Vision-and-language navigation(VLN) [4,28,25,35] is one of the most popular tasks of embodied AI, in which an embodied agent learns to navigation to a goal location in indoor environments following language instruction and given dynamic visual information. Anderson et al. [4] first propose a LSTM-based seq-to-seq model for navigation. For better understanding vision-and-language information, there are works working on vision-and-language pre-training [17,30,19,34,15] and model structures [19,6,13]. Reinforcement learning and navigation planning methods were also introduced into VLN to perform better action decisions [42,43,27,3]. Limited labeled data was another bottleneck to train a better model. To this end, Fried et al. [10] propose a speaker-follower model which can generate pseudo instructions for a sampled path by a trained speaker. Further, to mitigate the gap between seen and unseen environments, pre-exploration was proposed [42,40,11], which can learn and adapt to new environments after deployment. However, most current research ignores the practicality in real-life application scenarios, especially data privacy issues. Fu et al. [11] consider the implementation problem of pre-exploration and proposed environment-based pre-exploration, but they did not consider the privacy issue of training data. Also, we showed that environment-based pre-exploration might suffer from data scarcity and data bias.

[1] We conduct experiments on the English data of the RxR dataset.

2.2 Privacy-preserving Machine Learning

Over the years, researchers propose many methods [7,33,41,21] to address different data privacy problems [9,39,12,18] in machine learning. First, during the training stage, if the training data are from different parties, sharing their data with other parties might leads to privacy concerns. At the inference stage, there are multiple privacy attacks, especially in the scenario of Machine Learning as a Service (MLaaS), in which cloud providers offer machine inference hosted on the cloud [7]. For example, membership inference attack [39] can judge if a specific data sample exists in training data, model inversion attack [9,45] aims to infer training data given white-box or black-box access to the model. Also, in MLaaS, users might not be willing to directly upload their data to the cloud server [7]. Facing these privacy problems for training data and inference data, researchers propose many privacy-preserving methods, including federated learning, homomorphic encryption, differential privacy, etc [41,31,33,29]. However, most of their work focuses on single modality tasks and static data, and seldomly study the data privacy of embodied AI. In embodied AI tasks like vision-and-language navigation, the data contains more human-robot interaction and more complex private information, such as corresponding language-image pairs, dynamic visual information in the indoor environments. VLN also has a unique training stage, pre-exploration. Both of these may make the privacy problems and solutions for VLN more complex. In our work, we elaborate on privacy-preserving VLN training scenarios and propose a solution.

2.3 Federated Learning

Federated learning [41] is a technique that allows client models to train locally and then be sent to the central server for model aggregation. In this way, the clients do not need to send their sensitive data to any party. Thus the privacy of training data is protected. The first federated learning algorithm [41] uses weighted sum for aggregating clients' models. Later, researchers proposed different federated learning algorithms for heterogeneous data distribution and personalization [29,8,20,22]. Especially, Collins et al. [8] proposed to keep classification head locally for personalization. Compared with our framework, they were trying to solve the problem of label heterogeneity and learn a general data representation, and their setting does not have the difference between validation data and training data. Reddi et al. [37] summarized these first-order aggregation methods into one framework as FedOpt, whose server aggregation is:

$$w_{t+1} = \mathrm{SERVEROPT}(w_t, -\Delta w_t, \eta, t) \tag{1}$$

Where SERVEROPT is the aggregation algorithm, η is server learning rate.

Application wise, federated learning framework has been used on various tasks in computer vision [16,20] and natural language processing [32,23]. Recently, there are also some works for federated learning on multi-modal machine learning [1,46]. Zhao, et al. [46] try horizontal federated learning(FL), vertical FL, and Federated Transfer Learning on different multi-modal tasks and datasets, and [1] using semi-supervised FL to extract hidden representations of multi-modality. However, the tasks they discussed is not embodied agent for individual users. In vision-and-language navigation, the training paradigm is different from formerly discussed tasks, which has two different training objectives, training scenarios in two training stages. To solve this, we proposed a novel Federated Vision-and-Language Navigation(FedVLN) framework.

3 Privacy-preserving Vision-and-Language Navigation

3.1 Vision-and-Language Navigation (VLN)

The goal of the VLN task is to navigate from a given location and reach a destination following natural language instruction. The task can be formally defined as follow: given an language instruction $U = \{u_1, u_2, ..., u_n\}$. At each step, the agent will receive current visual information v_t as input. The agent will need to choose an action a_t at each step based on the instruction U, current/history visual information$\{v_\tau\}_{\tau=1}^{t}$, and history actions$\{a_\tau\}_{\tau=1}^{t-1}$. The agent's state, which consists of the agent's navigation history and current spatial location, will change according to the agent's action. The navigation terminates when the agent chooses a 'stop' action. The environments that contain labeled training data are seen environments. There are also unseen environments that do not have training data and are invisible during training.

VLN agents In general, VLN agents consist of a language encoding module to understand the instruction, a trajectory encoder to encode visual observation and actions, and a multimodal decision module to jointly process multi-modal information including encoded language information L_{enc}, visual information V_{enc}, and action information A_{enc} and predict the next action a_t:

$$L_{enc} = E_L(u_1, u_2, ..., u_n) \tag{2}$$

$$V_{enc}, A_{enc} = E_T(v_1, v_2, ..., v_t, a_1, a_2, ..., a_{t-1}) \tag{3}$$

$$a_t = M(L_{enc}, V_{enc}, A_{enc}) \tag{4}$$

Speaker-based data augmentation To tackle the problem of data scarcity, Fried et al. [10] propose a back-translation speaker model which can generate corresponding instructions U from the visual information and action sequence of sampled routes in the environment:

$$U = Speaker(v_1, v_2, ..., v_t, a_1, a_2, ..., a_t) \tag{5}$$

The speaker is trained by original labeled route-instruction pairs, which takes the visual and actions information of routes as input and predict the instructions. The generated pseudo instructions along with sampled routes can be the augmented training data for better agent learning.

Pre-exploration After training on seen environments and deploying on unseen environment, the agent can adapt to the new environment via pre-exploration [11,42,40]. There are different variants of pre-exploration includes self-imitation learning [42],

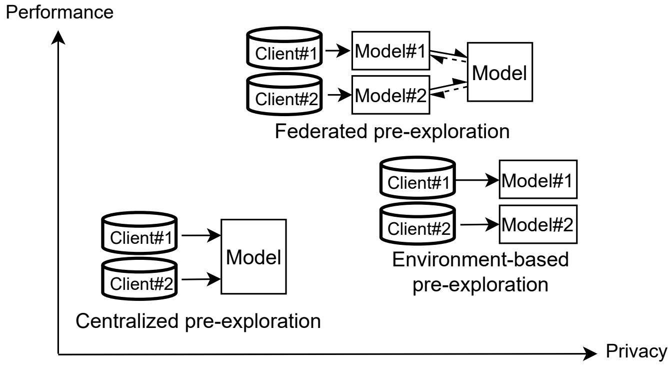


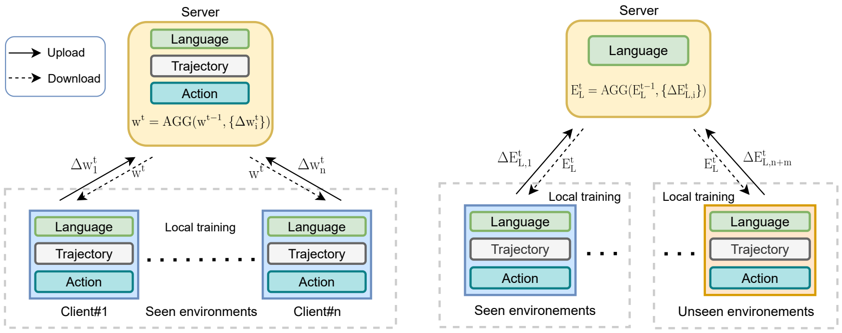
Fig. 2. Comparison between different pre-exploration strategies on performance-privacy trade-off. Federated pre-exploration achieves the best navigation performance while maintaining good inference data privacy protection.

graph-based methods [47,5], etc. In our work, we consider the paradigm that sampling routes $R^{'}$ from a new environment and generate instructions $I^{'}$ using the trained speaker mentioned before. Then the agent can be trained on the new environment using sampled routes and pseudo instructions($R^{'}, I^{'}$).

3.2 Privacy-preserving VLN

Considering the data may have sensitive information, users may have different levels of concern about the privacy of their data. In our work, we consider the case that the users do not want their data to be directly shared with the server (e.g., companies) and any other parties. Based on this, we define privacy-preserving vision-and-language navigation learning setting on two training stages: seen environment training and unseen environment pre-exploration. For seen environment training, including the training of navigation agent, speaker model and data augmentation process, no labeled data within the house environment will be directly shared with the server or any other client to prevent the leak of private information. And our primary purpose is to train a model that can generalize well on unseen environments. Thus, we need to utilize all the data indirectly to train one model.

For pre-exploration, the unlabeled data in unseen environments also can not be shared with others. However, the purpose in this stage is to adapt the model to a specific environment. Thus, training on data in one environment (environment-based pre-exploration) might not be a bad choice. In fact, our experiments show that environment-based pre-exploration performs better than centralized pre-exploration. Nevertheless, as elaborated in Sec. 4.2, we can indirectly utilize all the data in pre-exploration to boost the performance and preserve privacy. As in Fig. 2, we aim to achieve the best performance-privacy trade-off in pre-exploration.



(a) Decentralized Federated Training (b) Federated Pre-exploration

Fig. 3. The FedVLN framework. In the first stage (a), agents in seen environments will be trained on local data and upload the model updates to the server for aggregation (AGG), then download the global model from the server. In the second stage (b), all the agents in seen and unseen environments join the federated training. During local training, all the modules will be optimized, while only the language encoder will be uploaded/downloaded.

4 The FedVLN Approach

We propose a federated vision-and-language navigation framework (FedVLN) as shown in Fig. 3, in which user's data can be kept locally during both training and pre-exploration. In this section, we will introduce our FedVLN framework for two training stages: Decentralized Federated Training and Federated Pre-exploration. In decentralized federated training, each environment has a local agent, which will be trained on local data, then uploaded to the server. Then the global model on the server will be updated by the aggregation of local model updates and sent to all the environments. In federated pre-exploration, to enable the agent to both adapt to the new environment and maintain the ability to understand language, only the language encoder will be shared with the server after local training, instead of sharing the full model. All the agents from seen and unseen environments will join the federated pre-exploration process.

4.1 Decentralized Federated Training

Original training data When training on the original training data, we first divide the VLN dataset by environments. We treat each environment as a client, then assign a local navigation agent w_i^0 on each environment, which is initialized as the same as global navigation agent w^0. At each communication round between clients and server, a certain percentage of clients will be randomly selected for training, the local agent on each selected client will be trained for a certain number of epochs on their own data d_i:

$$w_i^t = \text{ClientUpdate}(w^{t-1}, d_i) \qquad (6)$$

Statistics	GT	Speaker
Length	29.58	21.89
Var(Length)	155.70	20.88
NoS	2.44	2.42
Var(NoS)	1.21	0.47

Table 1. Comparison between ground-truth (GT) and speaker generated instructions on seen validation. NoS means the average number of sentences.

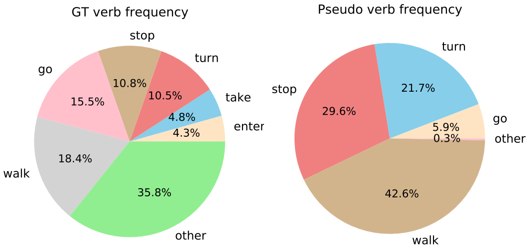


Fig. 4. Comparison of verb frequency between ground truth instructions and generated pseudo instructions.

Where ClientUpdate is the local training process. Then each selected client will send the update $\varDelta w_{i,t} = w_i^t - w^{t-1}$ of their model to the server, and the server will aggregate all the models with a server learning rate η:

$$w^t = w^{t-1} + \eta \sum_{i\in\phi_t} \frac{n_j}{\sum_{j\in\phi_t} n_j} \varDelta w_i^t \tag{7}$$

Here the weight of each local model $\frac{n_j}{\sum_{j\in\phi_t} n_j}$ is the proportion of the use's sample in the total training sample of this communication round.

Augmentation For data augmentation, we will assign each client a local speaker. Following the federated learning paradigm mentioned above and the training procedure of speaker from Sec. 3.1, at each communication round, each speaker from the selected clients will be trained on the labeled route-instruction pairs in its environment.

The best global model (according to BLUE score) during the training process will be sent to all clients. Each client can use the speaker model to generate pseudo instructions I_i^{aug} for sampled routes within the environment. Then the augmented training of the agent will also follow the federated training process mentioned above, except the local data will be the combination of original data and augmented data $\{(d_i, d_i^{aug})\}_{i=1}^n$.

Notice that during the whole training process, including original data training, speaker training, and augmented data training, no client share their data to other clients or the server. Thus the training data privacy is preserved.

4.2 Federated Pre-exploration

Pre-exploration allows the agent to explore the newly deployed environment and update itself based on the new information. From the perspective of data privacy, centralized pre-exploration is impractical here, since it assumes one navigation agent can get access to data from all the unseen environments. Fu et al.[11] proposed environment-based pre-exploration, which allows each agent to train on only one environment. Thus no data will be shared with other parties, and the data privacy in unseen environments is preserved. From the performance point of view, for centralized training, since the agent is trained on all the data from all the environments, it should have better generalizability. However, training in all the environments may hinder the agent from better adapting to one specific environment. For environment-based pre-exploration, the agent can focus on one specific environment, while the limited data amount and data bias in one environment may lead to a less generalized agent.

Furthermore, as shown in Table 1 and Fig. 4, we found that the instructions generated by the speaker are statistically significantly different from human-generated instructions. Moreover, the language pattern is much simpler than human language. Since current methods only use augmented data with speaker-generated instructions for training during pre-exploration, the agent might suffer from the huge distribution shift between instructions in augmented data and validation data, and can not understand instructions in validation data well. This problem could be even worse on environment-based pre-exploration since the data for one agent is of a smaller amount and from a single environment.

What is more, according to former research [44], the agent will perform better on seen paths or environments. Thus, the best solution is to maintain the generalizability to understand language and adapt to a specific visual environment. To this end, as in Fig. 3(b), we propose federated pre-exploration. In federated pre-exploration, The server will only maintain a global language encoder, which is initialized with the global encoder after decentralized federated VLN training. During each communication round, the server will send the global language encoder E^{t-1} to the selected clients. Then the selected clients will update its language encoder with E^{t-1}, and train the full agent on its local data:

$$E^t_{L,i}, E^t_{T,i}, M^t_i = \text{ClientUpdate}(E^{t-1}_{L,i}, E^{t-1}_{T,i}, M^{t-1}_i, \tau, \lambda) \tag{8}$$

After local training, the model will send only the language encoder $E^t_{L,i}$ to the server for aggregation as lines 9,11 in Alg. 1. In this way, the language encoder will be jointly updated on data from all the participated environments, thus being more generalized. Meanwhile, to further improve the generalizability of the language encoder, we randomly sample a fraction of seen environments at each communication round, where agents will also follow the training process above. The trajectory encoding module $E_{T,i}$ and multi-modal decision module M_i will keep training locally, which can help local agents adapt to their own environments. For validation, we used the local models after local training. The whole training procedure is in Alg. 1.

5 Experimental Setup

5.1 Datasets

We implement our federated learning framework on two datasets: Room-to-Room (R2R) [4] and Room-across-Room (RxR)(en) [28]. Both datasets are developed on the Matterport3D Simulator [4], a photorealistic 3D environment for embodied AI research.

Algorithm 1 Federated Pre-exploration

```
Parameters: Seen participation rate $r_1$, unseen participation rate $r_2$; local learning rate $\lambda$; server learning rate $\eta$; number of communication rounds $T$; number of seen environments $n$; number of unseen environments $m$; local training epochs $\tau$.
Initialize: $E^0_{L,i} = E^0_L$, $E^0_{T,i} = E^0_T$, $M^0_i = M^0$, for i in {1,2,...,n+m}
for t in [1,T] do
    Server sample $r_1 n$ seen environments and $r_2 m$ unseen environments as $\phi_t$
    Server send global language encoder to selected environments $E^{t-1}$
    for client in $\phi_t$ do
        Client update language encoder: $E^{t-1}_i = E^{t-1}$
        Client local training: $E^t_{L,i}, E^t_{T,i}, M^t_i =$ ClientUpdate$(E^{t-1}_{L,i}, E^{t-1}_{T,i}, M^{t-1}_i, \tau, \lambda)$
        Client upload delta of the language encoder $\varDelta E^t_i = E^t_i - E^{t-1}$ to the server
    end for
    Server update language encoder: $E^t_i = E^{t-1} + \eta \sum_{i \in \phi_t} \frac{n_j}{\sum_{j \in \phi_t} n_j} \varDelta E^t_i$
end for
```

R2R [4] is constructed by generating the shortest paths from sampled start and end points. Then collect three associated navigation instructions for each path using Amazon Mechanical Turk (AMT). The dataset contains 7,189 paths from 90 environments, and each path contains 3 instructions. The environments are split into 61 environments for training and seen validation, 11 for unseen validation, and 18 for testing. The environments in unseen validation and unseen test set do not appear in the training environments.

RxR [28] is proposed to mitigate shortcomings of former VLN datasets. Specifically, it is a large-scale dataset with multilingual instructions. It contains 16,522 paths and 126,069 instructions, among which 42,002 instructions are in English. RxR also ensures spatiotemporal alignments between instructions, visual percepts, and actions for agent training. The RxR dataset samples arbitrary paths from point to point (not necessarily shortest paths) to avoid data bias.

5.2 Evaluation Metrics

For both datasets, we report Success Rate (SR), Success Rate weighted by Path Length (SPL), Oracle Success Rate (OSR), and navigation Error (NE) as goal-oriented metrics. SR is calculated as the percentage of the agent stop within 3 meters from the end point. SPL [2] is defined as Success weighted by normalized inverse Path Length, which considers both navigation effectiveness and efficiency. OSR is the percentage of the agent visiting a point within 3 meters from the end point. NE is the average distance between the agent's final location and the end point. We also report Coverage weighted by Length Score (CLS) [26] and normalized Dynamic Time Warping (nDTW) [24] to validate the fidelity of navigation paths, which penalize the deviation from the reference path. SR and SPL are often considered as the primary metrics for VLN evaluation.

5.3 Baselines

Currently, we do not consider pre-training privacy, and VLN data pre-training infringes on data privacy. Thus, we choose two strong VLN baselines without VLN pre-training for experiments:

1. **Envdrop** [40]: the environment dropout model uses Bi-directional LSTM as the language encoder and attentive LSTM as the action decoder, a mixed learning objective of imitation learning and reinforcement learning.
2. **CLIP-ViL** [38]: the CLIP-ViL model adapts CLIP [36] visual encoder to improve vision and language encoding and matching for vision-and-language navigation.

5.4 Implementation Details

When training on seen environments, the total number of training steps of local models is the same as centralized training steps. At each communication round, we use the participation rate of $r = 0.2$, and train each local agent for $\tau = 5$ epochs on local data. For federated speaker training, we select the best model on seen validation data according to BLEU2 score to generate instructions.

During pre-exploration, we use the participation rate of $r_1 = 0.6$ for unseen environments. And we train each agent for $\tau_1 = 1$ epoch over unseen local dataset. When training across seen and unseen environments, we use the participation rate of $r_2 = 0.18$ for seen environments. To validate the effectiveness of our framework, for centralized pre-exploration, environment-based pre-exploration and federated pre-exploration, we use federated trained speaker to generate pseudo-instruction and train from federated trained navigation agent.

6 Results

6.1 Decentralized Federated Training

Seen environment training In Table 2 and Table 3, we report the results for seen environment training on R2R and RxR datasets for both baselines.

First, federated learning performs worse than centralized training on seen environments with an average of 2.43% SR gap. This is reasonable, as centralized training can easily overfit to the seen training data for better performance on seen environments, while for federated learning, because of the decentralized optimization over protected local data, the global model can not overfit to the seen environments as well as centralized training.

The performance on unseen environments tests the generalization ability of VLN models and is used for VLN evaluation. As seen in Table 2 and Table 3, on unseen environments, decentralized federated training achieves comparable results with centralized training, on both original data training and augmented data training across different VLN models. For example, FedEnvdrop performs better than Envdrop on R2R and nearly the same on RxR, and FedCLIP-ViL

Model	Val-Seen						Val-Unseen					
	NE↓	OSR↑	SPL↑	SR↑	CLS↑	nDTW↑	NE↓	OSR↑	SPL↑	SR↑	CLS↑	nDTW↑
Envdrop	4.71	65.6	53.2	56.1	66.8	55.0	5.87	52.7	40.9	44.5	57.1	42.3
FedEnvdrop	4.69	63.1	52.4	55.0	66.4	55.1	5.66	53.0	43.4	46.5	59.0	45.5
Envdrop$_{\text{aug}}$	3.81	73.1	56.5	62.4	65.4	55.2	5.37	61.7	40.9	50.0	50.5	38.1
FedEnvdrop$_{\text{aug}}$	4.00	69.3	53.8	61.8	70.9	60.9	5.41	56.9	46.3	49.8	60.3	47.1
CLIP-ViL	4.07	70.7	57.9	62.9	67.7	55.8	5.02	63.1	47.5	53.6	58.1	44.5
FedCLIP-ViL	4.28	67.2	55.8	60.4	65.7	53.3	4.91	61.9	47.6	53.4	57.9	44.4
CLIP-ViL$_{\text{aug}}$	3.52	75.0	61.7	66.8	69.3	58.6	4.59	67.4	50.7	57.0	59.2	46.4
FedCLIP-ViL$_{\text{aug}}$	4.13	69.8	58.2	62.6	67.3	56.7	4.80	65.9	49.8	56.3	59.2	46.1

Table 2. R2R Results of seen environment training. Envdrop is the centralized Envdrop model, and FedEnvdrop is the federated Envdrop model. Envdrop$_{\text{aug}}$ means the Envdrop model trained with augmented data. Our decentralized federated training outpeforms centralized training with Envdrop and achieves comparable results with CLIP-ViL on unseen environments.

Model	Val-Seen						Val-Unseen					
	NE↓	OSR↑	SPL↑	SR↑	CLS↑	nDTW↑	NE↓	OSR↑	SPL↑	SR↑	CLS↑	nDTW↑
Envdrop	7.97	51.6	38.0	40.7	58.8	54.0	8.42	45.1	31.8	35.0	55.6	50.6
FedEnvdrop	8.60	49.2	33.8	36.8	56.2	51.0	8.59	44.3	31.0	34.2	55.1	49.7
CLIP-ViL	6.92	56.8	42.3	46.5	60.0	56.2	7.38	50.6	34.9	39.5	55.6	51.2
FedCLIP-ViL	7.18	54.6	40.0	44.2	59.0	54.7	8.54	50.1	35.0	39.4	56.0	51.5

Table 3. RxR results of seen environment training. Decentralized federated training obtains comparable results with centralized training on unseen environments (e.g., only 0.1% SPL difference with the CLIP-ViL model).

obtains comparable results with CLIP-ViL on both R2R and RxR. Thus, in terms of generalization ability, our decentralized federated training method is comparable with centralized training while protecting the training data privacy.

Federated speaker *vs.* centralized speaker From Table 2, we notice that when training with augmented data, federated learning has an average gap of 0.45% on unseen SR compared with centralized training, which is slightly worse than results without augmented data. We suspect that it's because the federated speaker produce slightly worse augmented data than the centralized speaker. So we compare the BLEU score between federated speaker and centralized speaker in Table 4. Results show that federated speaker performs 1.15 worse than centralized speaker on seen validation on BLEU2 score, but is 0.65 better on unseen validation data, which is quite aligned with the navigation results. Since we do data augmentation on seen environments, federated speaker generates slightly lower-quality pseudo instructions.

To further validate this, we replace federated speaker with centralized speaker to generate the augmented data. Results are on Table 5, when trained with the

Speaker	Val-seen	Val-unseen
CLIP Cent	33.5	30.2
CLIP Fed	32.7	31.5
ResNet Cent	33.6	30.7
ResNet Fed	32.1	30.7

Table 4. Comparison of BLUE2 score between federated speaker and centralized speaker based on the CLIP encoder and the ResNet encoder on R2R.

Speaker	Envdrop	CLIP-ViT
Centralized	49.9	56.6
Federated	49.7	55.9

Table 5. Comparison of SR between two baselines on data generated by centralized speaker and federated speaker on R2R validation unseen averaged on two runs. Using centralized speaker improves the navigation performance.

same augmented data by centralized speaker, whose quality is still worse than original data, decentralized federated training also obtains comparable performance with centralized training. Thus, federated learning can train a generalized model on both original data and pseudo data.

6.2 Federated Pre-exploration

To validate the effectiveness of federated pre-exploration on unseen environments, we compare centralized pre-exploration, environment-based pre-exploration, and different federated pre-exploration methods: full model sharing (Fed-Full), sharing language encoder only (Fed-Lan), and sharing language encoder across seen and unseen environments (Fed-Lan+seen). Results are shown in Table 6.
Navigation performance For centralized pre-exploration and Fed-Full, in which one agent is optimized on data from all the environments, the agent can not adapt very well on one specific environment. For example, there is a gap of 3.85% on SR between centralized training and environment-based pre-exploration. When sharing only the language encoder during federated learning, the validation results improve significantly comparing with full model sharing (e.g. 4.6% on SR) since the agents can adapt to each environment better. Also, the generalization ability of language encoder is better than environment-based pre-exploration, since it is trained on more data across different environments. Thus sharing only the encoder in federated pre-exploration achieves better results comparing with environment-based pre-exploration. Federated pre-exploration with seen environments further improves the performance benefiting from human labeled data, and achieves around 1.8% SR improvement than environment-based pre-exploration.
Degree of privacy From the perspective of privacy preserving, environment-based pre-exploration is the best, where nothing in the unseen environments will be shared with others. Centralized training is clearly the worse, where all the observable data from unseen environments will be directly shared with the server. Federated pre-exploration only uploads the model updates to the server. Among federated methods, sharing only the language encoder protects data privacy better than full model sharing: it only shares the updates of language encoder, which accounts for only 24.6% of the parameters and keeps other modules completely

Model	Method	NE↓	OSR↑	SPL↑	SR↑	CLS↑	nDTW↑	Privacy↑
Envdrop	Centralized	3.81	75.8	61.2	64.6	71.4	64.6	0 - sharing data
	Env-based	3.63	77.2	62.0	65.9	72.3	66.5	**3** - no sharing
	Fed-Full	3.94	74.9	59.4	62.9	70.8	63.0	1 - model sharing (100%)
	Fed-Lan	3.56	78.4	63.0	66.7	**73.1**	67.1	2 - model sharing (24.6%)
	Fed-Lan+seen	**3.51**	**78.5**	**63.6**	**67.6**	**73.1**	**67.3**	2 - model sharing (24.6%)
CLIP-ViT	Centralized	3.66	75.4	61.7	66.1	70.1	62.5	0 - sharing data
	Env-based	3.45	78.0	65.2	69.2	72.5	65.8	**3** - no sharing
	Fed-Full	3.78	74.9	60.5	64.8	69.2	61.0	1 - model sharing (100%)
	Fed-Lan	3.27	**79.8**	66.4	70.1	**74.4**	**69.3**	2 - model sharing (24.6%)
	Fed-Lan+seen	**3.21**	**79.8**	**67.3**	**71.0**	**74.4**	68.7	2 - model sharing (24.6%)

Table 6. Comparison between different pre-exploration methods on R2R unseen validation. Fed-Full means full model sharing federated learning, Fed-Lan means sharing only language encoder in federated learning, Fed-Enc+seen means federated training with seen environments and sharing encoder only.

local. Training with seen environments will not make the training process less private, as seen environments already shared their parameter updates with the server in decentralized federated training process.

Overall, our federated pre-exploration method achieves a good performance-privacy trade-off. Centralized training is both worst in terms of navigation ability and privacy protection. Environment-based pre-exploration has the best privacy protection of unseen environment data. Federated pre-exploration achieves the best navigation results with little privacy cost by keeping all client data locally, and sharing only the language encoder model updates with the server.

7 Conclusion and Future Work

In this paper, we study the data privacy problems in vision-and-language navigation with respect to two learning scenarios: seen environment training and unseen environment pre-exploration. We propose a novel federated vision-and-language navigation (FedVLN) framework to preserve data privacy in both learning stages while maintaining comparable navigation performance. Furthermore, we present that federated pre-exploration can even outperform all previous pre-exploration methods and achieves the best performance-privacy trade-off. As the first work along this direction, our work does not consider adversarial attacks that can potentially recover data information from shared local model updates, and we believe future work can consider more embodied AI tasks and defend against privacy attacks for more data security.

Acknowledgement We thank Jing Gu, Eliana Stefani, Winson Chen, Yang Liu, Hao Tan, Pengchuan Zhang, and anonymous reviewers for their valuable feedback. This work is partially supported by the PI's UCSC start-up funding.

References

1. Federated learning for vision-and-language grounding problems. In: Proceedings of the AAAI Conference on Artificial Intelligence. pp. 11572–11579 (2020)
2. Anderson, P., Chang, A.X., Chaplot, D.S., Dosovitskiy, A., Gupta, S., Koltun, V., Kosecka, J., Malik, J., Mottaghi, R., Savva, M., Zamir, A.R.: On evaluation of embodied navigation agents. CoRR **abs/1807.06757** (2018), http://arxiv.org/abs/1807.06757
3. Anderson, P., Shrivastava, A., Parikh, D., Batra, D., Lee, S.: Chasing ghosts: Instruction following as bayesian state tracking. In: Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., Garnett, R. (eds.) Advances in Neural Information Processing Systems. vol. 32. Curran Associates, Inc. (2019)
4. Anderson, P., Wu, Q., Teney, D., Bruce, J., Johnson, M., Sünderhauf, N., Reid, I., Gould, S., van den Hengel, A.: Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2018)
5. Chen, K., Chen, J.K., Chuang, J., Vazquez, M., Savarese, S.: Topological planning with transformers for vision-and-language navigation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 11276–11286 (June 2021)
6. Chen, S., Guhur, P.L., Schmid, C., Laptev, I.: History aware multimodal transformer for vision-and-language navigation. In: NeurIPS (2021)
7. Chou, E., Beal, J., Levy, D., Yeung, S., Haque, A., Fei-Fei, L.: Faster cryptonets: Leveraging sparsity for real-world encrypted inference. CoRR (2018)
8. Collins, L., Hassani, H., Mokhtari, A., Shakkottai, S.: Exploiting shared representations for personalized federated learning. In: Meila, M., Zhang, T. (eds.) Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event. Proceedings of Machine Learning Research, vol. 139, pp. 2089–2099. PMLR (2021)
9. Fredrikson, M., Jha, S., Ristenpart, T.: Model inversion attacks that exploit confidence information and basic countermeasures. In: Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security. p. 1322–1333. CCS '15 (2015)
10. Fried, D., Hu, R., Cirik, V., Rohrbach, A., Andreas, J., Morency, L.P., Berg-Kirkpatrick, T., Saenko, K., Klein, D., Darrell, T.: Speaker-follower models for vision-and-language navigation. In: NeurIPS (2018)
11. Fu, T.J., Wang, X.E., Peterson, M.F., Grafton, S.T., Eckstein, M.P., Wang, W.Y.: Counterfactual vision-and-language navigation via adversarial path sampler. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 71–86 (2020)
12. Ganju, K., Wang, Q., Yang, W., Gunter, C.A., Borisov, N.: Property inference attacks on fully connected neural networks using permutation invariant representations. In: Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security. p. 619–633. CCS '18 (2018)
13. Gao, C., Chen, J., Liu, S., Wang, L., Zhang, Q., Wu, Q.: Room-and-object aware knowledge reasoning for remote embodied referring expression. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 3064–3073 (June 2021)

14. Gu, J., Stefani, E., Wu, Q., Thomason, J., Wang, X.: Vision-and-language navigation: A survey of tasks, methods, and future directions. In: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 7606–7623. Association for Computational Linguistics, Dublin, Ireland (May 2022). https://doi.org/10.18653/v1/2022.acl-long.524, https://aclanthology.org/2022.acl-long.524
15. Guhur, P.L., Tapaswi, M., Chen, S., Laptev, I., Schmid, C.: Airbert: In-domain pretraining for vision-and-language navigation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 1634–1643 (October 2021)
16. Guo, P., Wang, P., Zhou, J., Jiang, S., Patel, V.M.: Multi-institutional collaborations for improving deep learning-based magnetic resonance image reconstruction using federated learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 2423–2432 (June 2021)
17. Hao, W., Li, C., Li, X., Carin, L., Gao, J.: Towards learning a generic agent for vision-and-language navigation via pre-training. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2020)
18. Hisamoto, S., Post, M., Duh, K.: Membership Inference Attacks on Sequence-to-Sequence Models: Is My Data In Your Machine Translation System? Transactions of the Association for Computational Linguistics **8**, 49–63 (01 2020)
19. Hong, Y., Wu, Q., Qi, Y., Rodriguez-Opazo, C., Gould, S.: Vln bert: A recurrent vision-and-language bert for navigation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1643–1653 (June 2021)
20. Hsu, T.M.H., Qi, H., Brown, M.: Federated visual classification with real-world data distribution. In: Computer Vision – ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part X. p. 76–92 (2020)
21. Huang, Y., Song, Z., Chen, D., Li, K., Arora, S.: TextHide: Tackling data privacy in language understanding tasks. In: Findings of the Association for Computational Linguistics: EMNLP 2020. pp. 1368–1382 (Nov 2020)
22. Huang, Y., Chu, L., Zhou, Z., Wang, L., Liu, J., Pei, J., Zhang, Y.: Personalized cross-silo federated learning on non-iid data. Proceedings of the AAAI Conference on Artificial Intelligence **35**(9), 7865–7873 (May 2021), https://ojs.aaai.org/index.php/AAAI/article/view/16960
23. Huang, Z., Liu, F., Zou, Y.: Federated learning for spoken language understanding. In: Proceedings of the 28th International Conference on Computational Linguistics. pp. 3467–3478. International Committee on Computational Linguistics, Barcelona, Spain (Online) (Dec 2020)
24. Ilharco, G., Jain, V., Ku, A., Ie, E., Baldridge, J.: General evaluation for instruction conditioned navigation using dynamic time warping (2019)
25. Jain, V., Magalhaes, G., Ku, A., Vaswani, A., Ie, E., Baldridge, J.: Stay on the path: Instruction fidelity in vision-and-language navigation. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. pp. 1862–1872. Association for Computational Linguistics (Jul 2019)
26. Jain, V., Magalhaes, G., Ku, A., Vaswani, A., Ie, E., Baldridge, J.: Stay on the path: Instruction fidelity in vision-and-language navigation. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. pp. 1862–1872. Association for Computational Linguistics, Florence, Italy (Jul 2019)
27. Krantz, J., Gokaslan, A., Batra, D., Lee, S., Maksymets, O.: Waypoint models for instruction-guided navigation in continuous environments. In: Proceedings of

the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 15162–15171 (October 2021)

28. Ku, A., Anderson, P., Patel, R., Ie, E., Baldridge, J.: Room-across-room: Multilingual vision-and-language navigation with dense spatiotemporal grounding. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 4392–4412 (Nov 2020)
29. Li, Q., He, B., Song, D.: Model-contrastive federated learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 10713–10722 (June 2021)
30. Li, X., Yin, X., Li, C., Zhang, P., Hu, X., Zhang, L., Wang, L., Hu, H., Dong, L., Wei, F., Choi, Y., Gao, J.: Oscar: Object-semantics aligned pre-training for vision-language tasks. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J. (eds.) Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XXX. Lecture Notes in Computer Science, vol. 12375, pp. 121–137. Springer (2020)
31. Lou, Q., Jiang, L.: She: A fast and accurate deep neural network for encrypted data. In: Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., Garnett, R. (eds.) Advances in Neural Information Processing Systems. vol. 32 (2019)
32. Lu, Y., Huang, C., Zhan, H., Zhuang, Y.: Federated natural language generation for personalized dialogue system (2021)
33. Papernot, N., Song, S., Mironov, I., Raghunathan, A., Talwar, K., Erlingsson, Ú.: Scalable private learning with PATE. In: 6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings (2018)
34. Qi, Y., Pan, Z., Hong, Y., Yang, M.H., van den Hengel, A., Wu, Q.: The road to know-where: An object-and-room informed sequential bert for indoor vision-language navigation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 1655–1664 (October 2021)
35. Qi, Y., Wu, Q., Anderson, P., Wang, X., Wang, W.Y., Shen, C., van den Hengel, A.: REVERIE: remote embodied visual referring expression in real indoor environments. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020. pp. 9979–9988. Computer Vision Foundation / IEEE (2020)
36. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., Sutskever, I.: Learning transferable visual models from natural language supervision. In: Meila, M., Zhang, T. (eds.) Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event. Proceedings of Machine Learning Research, vol. 139, pp. 8748–8763. PMLR (2021)
37. Reddi, S.J., Charles, Z., Zaheer, M., Garrett, Z., Rush, K., Konečný, J., Kumar, S., McMahan, H.B.: Adaptive federated optimization. In: International Conference on Learning Representations (2021)
38. Shen, S., Li, L.H., Tan, H., Bansal, M., Rohrbach, A., Chang, K., Yao, Z., Keutzer, K.: How much can CLIP benefit vision-and-language tasks? CoRR **abs/2107.06383** (2021), `https://arxiv.org/abs/2107.06383`
39. Shokri, R., Stronati, M., Song, C., Shmatikov, V.: Membership inference attacks against machine learning models. In: 2017 IEEE Symposium on Security and Privacy (SP). pp. 3–18 (2017)

40. Tan, H., Yu, L., Bansal, M.: Learning to navigate unseen environments: Back translation with environmental dropout. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers). pp. 2610–2621 (Jun 2019)
41. Vanhaesebrouck, P., Bellet, A., Tommasi, M.: Decentralized collaborative learning of personalized models over networks. In: Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, AISTATS 2017, 20-22 April 2017, Fort Lauderdale, FL, USA. pp. 509–517. Proceedings of Machine Learning Research (2017)
42. Wang, X., Huang, Q., Celikyilmaz, A., Gao, J., Shen, D., Wang, Y.F., Wang, W.Y., Zhang, L.: Reinforced cross-modal matching and self-supervised imitation learning for vision-language navigation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2019)
43. Wang, X., Xiong, W., Wang, H., Wang, W.Y.: Look before you leap: Bridging model-free and model-based reinforcement learning for planned-ahead vision-and-language navigation. In: Proceedings of the European Conference on Computer Vision (ECCV) (September 2018)
44. Zhang, Y., Tan, H., Bansal, M.: Diagnosing the environment bias in vision-and-language navigation. In: Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence. IJCAI'20 (2021)
45. Zhang, Y., Jia, R., Pei, H., Wang, W., Li, B., Song, D.: The secret revealer: Generative model-inversion attacks against deep neural networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2020)
46. Zhao, Y., Barnaghi, P., Haddadi, H.: Multimodal federated learning on iot data (2022)
47. Zhou, X., Liu, W., Mu, Y.: Rethinking the spatial route prior in vision-and-language navigation. CoRR **abs/2110.05728** (2021)