FedNASP: Federated Vision-Language Navigation with Adaptive Step-wise Personalization

Qingqian Yang[1], Hao Wang[1(✉)]
, Stevens Institute of Technology, Hoboken, NJ, USA
Sai Qian Zhang[2], Jian Li[3], Yang Hua[4], Miao Pan[5], Tao Song[6], Zhengwei Qi[6], and Haibing Guan[6]

No Institute Given

Abstract. Federated learning (FL) protects sensitive (Vision-Language Navigation) VLN data without centralizing trajectories or instructions, but severe non-IID environments make personalized FL (pFL) necessary. Moreover, VLN poses several coupled challenges for personalized federated learning, including environment heterogeneity, multimodal language-vision fusion, and long-horizon navigation with time-varying decision contexts. To address these challenges, we propose FedNASP, a step-wise personalized federated learning framework for VLN. The key idea is to dynamically calibrate personalization strength along a navigation trajectory. Specifically, we introduce a lightweight Step-wise Personalized Modulator (SPM) that predicts personalization strength at each navigation step. We further design a structure-aware adapter-based personalized prefix injection mechanism that enables client-specific grounding while keeping the backbone shared across clients. Experiments on three representative datasets show that FedNASP consistently outperforms state-of-the-art federated VLN methods under substantial cross-client heterogeneity. Compared with the non-centralized baselines, FedNASP improves Remote Grounding Success on REVERIE by 13.0% and Success Rate on CVDN by 22.6%. Extensive ablation studies and visualizations further validate the effectiveness of adaptive step-wise personalization for federated VLN. Code is available at: https://github.com/IntelliSys-Lab/FedNASP.git

Keywords: Vision-Language Navigation · Personalized Federated Learning· Embodied AI

1 Introduction

Vision-Language Navigation (VLN) is widely regarded as a key challenge in embodied intelligence, since it evaluates whether an agent can ground natural-language instructions in egocentric visual observations and reliably execute long-horizon navigation in complex 3D environments [1, 8, 27]. Achieving robust VLN performance typically relies on large-scale training data because the agent must

Fig. 1: Motivation of step-wise personalized federated VLN. (a) Client-induced heterogeneity: different houses exhibit distinct visual statistics, and users describe routes with diverse instruction styles, yielding severe non-IID data across clients in federated training. **(b)** Long-horizon, time-varying decision contexts: within a single episode, the agent may alternate between uncertain steps (e.g., unfamiliar turns) and confident steps (e.g., familiar hallways), motivating step-wise adaptation of personalization strength rather than a fixed per-client or per-episode strategy.

perform complex multimodal reasoning, grounding ambiguous language into visual observations and sequential decisions [6]. However, real-world VLN data are often collected from private homes or offices: trajectories, panoramic observations, and user-written instructions can reveal sensitive information such as house layouts, objects present, and even daily routines. This privacy barrier makes the conventional assumption of centralizing all user data on a server unrealistic, and recent studies have highlighted that privacy is a largely overlooked but critical gap between VLN research and practical deployment in personal spaces [12, 27].

Federated Learning (FL) provides a natural training solution for this setting: each VLN agent (or each environment) keeps raw data locally and participates by communicating only model updates. FedVLN [27] demonstrates the feasibility of decentralized VLN training, yet a single shared model obtained via standard aggregation (*e.g.* FedAvg [11]) can be a poor fit when clients correspond to distinct environments. As Fig. 1 illustrates, treating each house agent as a client yields substantial cross-client shifts in both environmental layout and appearance, and instruction style; under such severe non-IID conditions, naively averaging updates often produces a compromised policy that is not optimal for any individual client, particularly for VLN where precise instruction grounding and reliable exploration are essential.

These observations motivate personalized Federated Learning (pFL) for VLN. pFedNavi [23] is the first pFL method for VLN. However, its layer selection strategy and parameter-wise personalization can be suboptimal for VLN, since it incurs high optimization overhead and fails to deal with multi-dimensional

challenges in VLN. This is mainly because multiple difficulty sources in VLN are tightly coupled. **(i) Severe, multi-dimensional client heterogeneity.** In federated VLN, each client naturally corresponds to a distinct house/environment. As illustrated in Fig. 1a, different houses exhibit markedly different visual statistics and spatial structures. In practice, this heterogeneity is not a single-factor shift but manifests simultaneously across complementary dimensions: environments vary in spatial extent, layout organization, and navigation topology, inducing navigation graphs with diverse sizes and connectivity patterns and consequently shaping distinct trajectory distributions. As a result, clients operate under heterogeneous observation spaces and navigation structures, making the personalization difficult to generalize across environments. **(ii) Multimodal fusion amplifies distribution shifts.** VLN policies rely on cross-modal fusion to align language with visual context for action selection. Under client shifts, maintaining robust language–vision alignment becomes harder; personalization that improves one modality can inadvertently perturb cross-modal grounding. **(iii) Long-horizon navigation with time-varying decision contexts.** VLN is a sequential decision problem where uncertainty is non-uniform along an episode. Fig. 1b illustrates that decision difficulty is *non-uniform* within an episode: some steps are almost deterministic given the memory and observation, whereas others require resolving ambiguity among multiple candidates. This makes fixed personalization brittle: applying a fixed personalization level throughout an episode can be overly aggressive when shared knowledge should dominate, yet insufficient when local, step-specific disambiguation is required. Moreover, because early mistakes compound over time, miscalibrated personalization at a few key steps can disproportionately harm overall success.

We address these challenges with `FedNASP`, a *step-wise* personalized federated learning framework for VLN. `FedNASP` dynamically calibrates personalization strength along a navigation trajectory to match time-varying decision contexts while preserving globally transferable navigation knowledge. Specifically, a lightweight step-wise prefix modulator (SPM) predicts step-wise modulation scalars from the current navigation state and history memory. To support structure-aware personalization in multimodal VLN, `FedNASP` employs adapter-based personalized prefix injection (SPI) on selected attention layers, whose influence is modulated by SPM. Overall, `FedNASP` balances shared knowledge and client-specific grounding across steps, enabling robust federated VLN under severe heterogeneity and long-horizon uncertainty. Building on this insight, we make the following contributions:

- **Step-wise personalized federated VLN.** We propose `FedNASP`, a step-wise personalized federated learning framework for VLN that dynamically calibrates personalization strength along a navigation trajectory to handle long-horizon and time-varying decision contexts.
- **Structure-aware personalization.** We empirically analyze the personalization sensitivity of different functional modules in VLN models, and design a structure-aware personalized prefix injection (SPI) mechanism to enable module-aware client-specific adaptation.

– **Empirical validation on federated VLN benchmarks.** We conduct extensive experiments on representative VLN settings, including object-grounding VLN (REVERIE [15]), dialog-based VLN (CVDN) [21], and standard navigation VLN (R2R [1]), under federated training with severe cross-client heterogeneity. `FedNASP` consistently outperforms strong federated VLN and personalized FL baselines. Compared with the non-centralized baselines, `FedNASP` improves Remote Grounding Success on REVERIE by 13.0% and Success Rate on CVDN by 22.6%. Extensive ablations and trajectory-level visualizations further validate the effectiveness of adaptive step-wise personalization.

2 Related work

VLN Tasks: Heterogeneity and Rising Task Complexity. Recent VLN tasks increasingly reflect real-world deployment, where diverse environments and instruction sources naturally amplify difficulty. Early action-directed tasks (*e.g.* R2R [1], RxR [9]) built on Matterport3D [3] operate on a *fixed topology* (panoramic viewpoints with pre-defined links), yet still exhibit strong cross-house variation in layout, object distributions, and visual statistics. Goal-directed tasks (*e.g.* REVERIE [15], SOON [28]) raise complexity by requiring *object grounding*, making language–vision alignment more environment-sensitive. Interactive settings (*e.g.* HANNA [13], CVDN [21]) add *multi-turn dialog*, where varied speaking styles and incremental updates intensify instruction heterogeneity. Beyond nav-graphs, VLN-CE [8] and Robo-VLN [7] evaluate *continuous environments* (*e.g.* Habitat [16]/Matterport3D [3]) with low-level control, where small execution biases and local layout differences can significantly affect outcomes. Task-oriented embodied settings (*e.g.* ALFRED [17], TEACh [14], DialFRED [4]) further add interaction/manipulation, stressing long-horizon planning under multimodal uncertainty. Together, these tasks show how environment and instruction heterogeneity compounds into long-horizon multimodal complexity.

VLN Models: Stronger Multimodal Pretraining and Explicit Reasoning. Early VLN agents are largely seq2seq, pairing an instruction encoder with an attention-based policy decoder, and are improved via data augmentation and environment dropout for better generalization [20]. Recent approaches increasingly adopt transformer-based multimodal backbones with cross-modal attention, where *pretraining + fine-tuning* is dominant; e.g., AirBERT performs in-domain VLN pretraining on large-scale BnB/Airbnb data to reduce domain shift and improve transfer [5]. In parallel, LLM/VLM-based methods introduce higher-level planning and reasoning (NavGPT-2 [26], UniGoal [24]). Recent analyses further note that physical and visual disparities in embodied deployment can substantially degrade performance, underscoring the need for robustness under heterogeneous conditions [22]. Overall, model advances mirror task evolution: coping with heterogeneous environments and instructions requires adaptive and robust long-horizon multimodal decision-making.

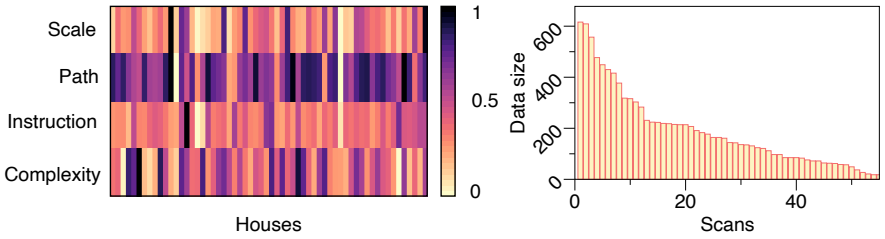

Fig. 2: House-level heterogeneity on REVERIE [15]. We visualize house-level statistics along four dimensions: *Instruction*, measured by the average instruction length; *Path*, measured by the variance of navigation trajectory lengths within each house; *Scale*, measured by the number of rooms, indicating the house size; and *Complexity*, measured by the number of nodes in the induced navigation graph, indicating how many navigation states and decisions the agent needs to encounter. Left: normalized statistics for each house/scan (min–max scaled to $[0, 1]$ across houses), where darker colors indicate smaller values. Right: the distribution of environment size shows a long-tail pattern. Together, these results confirm substantial non-IID heterogeneity across environments in REVERIE, motivating personalized learning.

Personalized Federated VLN. Real-world VLN data often reside in private homes/offices, making centralized training impractical. Federated learning (FL) enables collaborative training by keeping raw data local and exchanging only model updates; however, FedVLN [12, 27] shows that standard aggregation (*e.g.* FedAvg) struggles under the severe non-IID setting of VLN, where each client corresponds to a distinct house with shifts in layout/appearance and instruction style. As a result, naive averaging can produce a "compromise" policy that fits no client well, motivating personalized FL for VLN to balance globally transferable navigation knowledge with client-specific adaptation. Early pFL efforts such as pFedNavi [23] take steps in this direction, but can incur high overhead and be suboptimal when personalization is overly static or parameter-granular, limiting robustness to VLN's coupled heterogeneity (visual/statistical shifts, language–vision misalignment, and long-horizon decision dynamics).

3 Motivating Step-wise Personalized FL for VLN

While FL protects privacy and enables knowledge sharing, *vanilla* federated VLN (e.g., FedVLN [27]) that aggregates client updates into a single global policy often underperforms under VLN's severe non-IID conditions. As Fig. 2 shows, at the house level, clients exhibit substantial heterogeneity along multiple dimensions, including instruction statistics, path variability, environment scale, and navigation complexity. Such heterogeneity changes both the observations and the decision structure: different houses vary in visual content (*e.g.* objects, textures) and in spatial layout, which in turn reshapes the set of feasible actions and the local ambiguity at each step. Because VLN decisions are explicitly conditioned on the instruction, these environment shifts also alter how the same

language cue should be grounded into step-level action choices. Consequently, the instruction-to-action grounding becomes client-dependent, motivating the need for personalized federated learning (pFL) in VLN.

However, directly applying existing pFL techniques—most of which are developed for unimodal classification—is still insufficient for VLN. Most prior pFL methods assume that personalization can be captured by *static* client-level adaptation [2, 10, 18, 19, 25] that remains fixed throughout an episode. This assumption is often violated in VLN because several challenges are *coupled*: severe environment-induced heterogeneity, multimodal fusion, and long-horizon trajectories exhibit step-varying uncertainty. Under this joint effect, static episode-level personalization can easily become miscalibrated—over-adapting when shared knowledge should dominate and under-adapting when local, step-specific cues are critical.

We address these challenges with an adaptive step-wise pFL framework that enables the agent to determine an appropriate personalization strategy based on the current environment and history memory (Fig. 3).

Design Objectives: 1) **Heterogeneity-robust personalization:** apply client-specific personalization to diverse instruction styles and environments while maintaining generalization. 2) **Step-wise long-horizon adaptation:** dynamically adjust personalization strength during each navigation episode; 3) **Multimodal grounding preservation:** enable structure-aware personalization without sacrificing multimodal alignment.

Key Challenges: These objectives translate into three key challenges that directly motivate our design components:

- **Non-IID drift makes naive aggregation suboptimal.** With extreme cross-environment and cross-instruction heterogeneity, FedAvg-style averaging can produce a global model that is not optimal for any particular client, and local updates can be inconsistent across rounds.
- **Time-varying personalization demand within a navigation episode.** VLN is a sequential, long-horizon decision process: an agent's grounding quality can vary substantially across steps due to unfamiliar states or imperfect history memory. Consequently, a fixed personalization strategy within an episode is often insufficient when adaptation is needed, or overly aggressive when shared general knowledge should dominate.
- **Where to personalize matters: module selection is performance-critical.** VLN models are tightly coupled across functional components, yet these components are not equally sensitive to client heterogeneity. Personalizing the wrong modules can yield negligible gains and may even degrade performance. Conversely, relying too heavily on a global model can under-adapt to environment-specific patterns, limiting performance on heterogeneous clients.

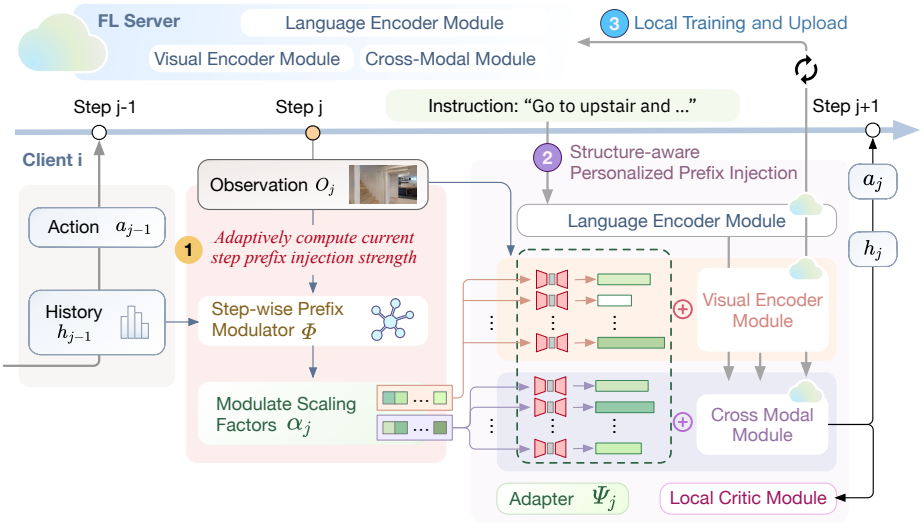

Fig. 3: Workflow of FedNASP from client i's perspective. (1) SPM predicts step-wise scaling factors α_t from the current state and history memory. (2) SPI uses client-local adapters Ψ_i to generate additive projection deltas for selected attention modules, with injection strength modulated by α_t. (3) Clients train locally, upload only backbone updates for FedAvg aggregation, and keep SPM/SPI parameters local.

4 The Design of FedNASP

4.1 Overview

We propose a step-wise personalized federated learning framework for VLN, where each client performs adaptive personalization during a navigation episode. **Step ① Adaptively predict the modulate scaling factors via a Step-wise Personalization Modulator (SPM):** At each navigation step, a lightweight modulator predicts a sequence of scaling factors based on the current state and history memory to control the injection strength of personalized prefixes. **Step ② Structure-aware Personalized Prefix Injection (SPI) with modulated strength:** For selected attention modules, we perform adapter-based SPI, where lightweight adapters produce additive projection deltas and the SPM outputs modulate their injection strength at each step. **Step ③ Local training & communication:** Each client locally updates its personalized parameters (adapter and SPM) together with the backbone using standard VLN objectives, and uploads only backbone updates; the server aggregates and broadcasts the global backbone via FedAvg, while personalized parameters are kept local for efficient and privacy-friendly personalization.

4.2 Formulating pFL for VLN

Personalized federated objective. Client i ($i \in \{1, \ldots, N\}$) hosts a VLN agent in environment Env_i and owns a private dataset $\mathcal{D}_i$ of instruction–trajectory pairs (I, τ). The server maintains a shared global backbone $\boldsymbol{\Theta}_g$, which is periodically broadcast to clients. In our setting, at communication round t, each client i maintains local personalized model: $\Theta_i^t \triangleq (\Theta_g^t,\ \Psi_i^t,\ \Phi_i^t)$, where Ψ_i^t denotes adapter parameters, and Φ_i^t denotes the SPM parameters. When reinforcement learning is enabled, client i additionally maintains a local critic with parameters $\boldsymbol{\Omega}_i$. Overall, the training objective is

$$\min_{\{\boldsymbol{\Theta}_i, \boldsymbol{\Omega}_i\}_{i=1}^N} \sum_{i=1}^{N} \mathbb{E}_{(I,\tau)\sim\mathcal{D}_i} \Big[\mathcal{L}_{\text{IL}}(I, \tau; \Theta_i) + \mathcal{L}_{\text{RL}}(I, \tau; \Theta_i, \Omega_i) \Big],$$

where $\mathcal{L}_{\text{IL}}$ denotes the imitation loss, and $\mathcal{L}_{\text{RL}}$ (when used) is an actor–critic objective that depends on the local critic $\boldsymbol{\Omega}_i$.

Step-wise Personalization Modulator (SPM) Since the decision context and environment-specific ambiguity vary across steps in a long-horizon VLN episode, a fixed personalization strength is suboptimal and should be adjusted dynamically step by step. To achieve this, we employ SPM to learn the relationship between the current observation, history memory, and the appropriate personalization strategy. Intuitively, SPM learns to assess when stronger personalization is beneficial, enabling it to select a reasonable degree of personalization for each step.

SPM Input Process. Intuitively, step-wise personalization should depend on two complementary cues: (1) *what the agent has accumulated so far* and (2) *what the agent currently observes*. The history state $h_{i,j-1}$ is the decoder (recurrent) hidden state, encoding trajectory memory, instruction-grounding progress, and prior decisions. The current observation feature $O_{i,j}$ captures the immediate visual context and local ambiguity at step j. Using only these two signals keeps SPM lightweight and directly tied to the online decision process, without introducing extra expensive statistics. Therefore, the input of SPM can be formulated as $z_{i,j} = \big[h_{i,j-1}; O_{i,j}\big]$, where $[\cdot;\cdot]$ denotes feature concatenation.

SPM Structure and Output. Given the compact step state $z_{i,j}$, SPM adopts a lightweight two-branch MLP: a history branch for $h_{i,j-1}$ and an observation branch for $O_{i,j}$. This decomposition is intentional: the history branch captures decision inertia (what has been inferred and executed so far), while the observation branch captures current local evidence (what is immediately visible now). Conditioned on the current step state, their fusion enables SPM to decide how much the agent should rely on SPI versus the globally shared backbone.

Formally, SPM maps $z_{i,j}$ to block-wise scaling factors over the selected personalized attention modules:

$$\boldsymbol{\alpha}_{i,j} = \sigma(g_{\Phi_i}(z_{i,j})) \in (0,1)^{|\mathcal{B}|}, \tag{1}$$

where Φ_i are client-local SPM parameters, and $\mathcal{B}$ denotes the set of personalized attention blocks (i.e., target layers where additive prefixes are injected). Each

Table 1: Which module should be personalized? This study was conducted on REVERIE using VilBERT. For each client, we initialize its training model by replacing the shared modules with a global model checkpoint Θ_g (trained from FedAvg) while keeping the remaining local modules as the personalized modules. We then perform the local training process and evaluate client-level performance. We report the mean client performance: Success Rate (SR) and Success weighted by Path Length (SPL). The columns indicate which modules are kept local.

Single Module	SR	SPL	Module combinations	SR	SPL
All global	27.83	22.08	Cross + Language	33.14	27.97
Language	13.23	11.66	Cross + Visual	36.59	29.79
Visual	17.48	15.37	Cross + Critic	33.92	27.71
Cross-modal	33.25	28.52	Cross + Visual + Critic	37.35	31.21

$\alpha_{i,j}^{(b)}$ corresponds to one block $b \in \mathcal{B}$. Importantly, a block may contain multiple internal additive branches (e.g., separate query- and key/value-prefix branches), which are jointly modulated by the same $\alpha_{i,j}^{(b)}$. The sigmoid σ constrains each factor to a stable bounded range, so each module is softly modulated rather than hard-switched, which improves optimization stability in long-horizon rollout.

4.3 Structure-aware Personalized Prefix Injection (SPI)

Why structure-aware personalization. VLN models differ from standard unimodal networks due to multimodal grounding, and different components exhibit different degrees of environment sensitivity. Accordingly, FedNASP decomposes a VLN agent into four functional modules and personalizes them differently. To identify where personalization is most effective, we conduct a module sensitivity study in Tab. 1: for each client, we keep a chosen subset of components local while replacing the rest with the FedAvg global checkpoint Θ_g, followed by local training and evaluation.

- **Language encoder module.** This module encodes instructions into contextualized language representations and is largely client-invariant. Tab. 1 shows that keeping it local degrades performance and provides no additional benefit when combined with cross-modal personalization, so we keep it globally shared (also reducing overhead).
- **Visual encoder module.** This module encodes panoramic observations and candidate-view features and is sensitive to environment-specific visual statistics. While visual-only personalization is ineffective in Tab. 1, jointly personalizing visual and cross-modal modules improves performance; thus we personalize the visual encoder in a coordinated manner.
- **Cross-modal fusion module.** This module performs language–vision alignment and is highly heterogeneity-sensitive. Tab. 1 shows clear gains when keeping it local, and it is consistently involved in the best configurations; therefore, we treat it as the primary personalization target.

Algorithm 1: Client-side Local Training Process

Input: Global backbone Θ_g^t, local SPM Φ_i, local adapter Ψ_i, local data $\mathcal{D}_i$
Output: Client backbone update $\Delta\Theta_i^t$, updated local (Φ_i, Ψ_i)
Initialize local backbone: $\Theta_i \leftarrow \Theta_g^t$;
for $e = 1, \ldots, E$ **do**
  foreach *local rollout batch* **do**
    Initialize history state $h_{i,0}$;
    for $j = 1, \ldots, L$ **do**
      Compute observation summary $O_{i,j}$ from current candidates;
      $z_{i,j} \leftarrow [h_{i,j-1}; O_{i,j}]$ $\alpha_{i,j} \leftarrow \sigma\big(g_{\Phi_i}(z_{i,j})\big)$;
      Run policy forward with $(\Theta_i, \Psi_i, \Phi_i, \alpha_{i,j})$ to get logits and next state $h_{i,j}$ Select action $a_{i,j}$;
    Compute task loss L_{task} (IL + (RL));
    Update $(\Theta_i, \Phi_i, \Psi_i)$ by backprop on L_{task};
$\Delta\Theta_i^t \leftarrow \Theta_i - \Theta_g^t$ Upload $\Delta\Theta_i^t$; keep (Φ_i, Ψ_i) local;

– **Critic module.** Since value estimation depends on environment-specific dynamics, we keep the critic local for stability; Tab. 1 shows that including a local critic does not hurt and can be beneficial when combined with other personalized modules.

Structure-aware Adapter-based Personalized Prefix Injection. We implement SPI via lightweight, structure-aware adapters that produce *additive* ΔQKV terms for attention projections. For each personalization-sensitive attention block $b \in \mathcal{B}$ (within the selected modules discussed above), client i maintains a local adapter $\Psi_i^{(b)}$. At step j, given the block input $x_{i,j}^{(b)}$, the adapter outputs additive terms for the query/key/value projections:

$$(\Delta Q_{i,j}^{(b)},\ \Delta K_{i,j}^{(b)},\ \Delta V_{i,j}^{(b)}) = \Psi_i^{(b)}(x_{i,j}^{(b)}).$$

We inject these terms by shifting the original projections:

$$\tilde{Q} = Q + \alpha_{i,j}^{(b)} \Delta Q, \quad \tilde{K} = K + \alpha_{i,j}^{(b)} \Delta K, \quad \tilde{V} = V + \alpha_{i,j}^{(b)} \Delta V,$$

where the step-wise *block-wise* scaling factor $\alpha_{i,j}^{(b)}$ (Eq. (1)) controls the effective injection strength. Importantly, a block may contain multiple internal additive injection branches, which are jointly modulated by the same $\alpha_{i,j}^{(b)}$. The attention output of block b is then computed using $(\tilde{Q}, \tilde{K}, \tilde{V})$, while blocks outside $\mathcal{B}$ remain unchanged. Overall, SPI realizes prefix-style personalization through adapter-generated ΔQKV terms, enabling structure-aligned local adaptation.

4.4 Local Training and Communication

Optimization objective. As Algorithm 1 shows, at global round t, each selected client $i \in \mathcal{S}_t$ optimizes a local VLN objective with step-wise personalization:

$$\min_{\Theta_i,\ \Phi_i,\ \Psi_i} \ \mathbb{E}_{\tau\sim\mathcal{D}_i}\left[\sum_{j=1}^{L} \ell_{\mathrm{IL}}\big(\pi_{\Theta_i,\Phi_i,\Psi_i}(s_{i,j}), a^{\star}_{i,j}\big) + \ell_{\mathrm{RL}}\big(\pi_{\Theta_i,\Phi_i,\Psi_i}(s_{i,j})\big)\right],$$

where $\pi_{\Theta_i,\Phi_i,\Psi_i}$ denotes the VLN policy parameterized by $(\Theta_i, \Phi_i, \Psi_i)$; $s_{i,j}$ is the navigation state at step j of trajectory τ from client i, and $a^{\star}_{i,j}$ is the teacher action (RL is optional).

Federated aggregation. After local training, client i uploads only its backbone update $\Delta\Theta^t_i$, while keeping the personalized parameters (Φ_i, Ψ_i) local. The server updates the global backbone via FedAvg: $\Theta^{t+1}_g \leftarrow \Theta^t_g + \sum_{i\in\mathcal{S}_t} \frac{|\mathcal{D}_i|}{\sum_{k\in\mathcal{S}_t}|\mathcal{D}_k|}\,\Delta\Theta^t_i$.

5 Evaluation

FL settings. All experiments are conducted on a single machine with NVIDIA RTX 3090 GPUs. We treat each agent deployed in a building (i.e., one environment/scan) as an FL client. In each FL round, we randomly sample a fraction $C = 0.2$ of clients and perform $E = 3$ local epochs. We run $R = 400$ communication rounds for REVERIE [15] and $R = 2000$ rounds for CVDN [21]. Each result is obtained as the average over five independent runs.

REVERIE Dataset. REVERIE [15] is an object-oriented VLN benchmark in photo-realistic indoor environments. Given an instruction, the agent must navigate and stop at a viewpoint from which the target object is visible, requiring both long-horizon navigation and fine-grained language–vision grounding. More implementation details are in the supplementary.

- **Model:** We adopt a transformer-based VLN agent with a ViLBERT-style backbone and initialize it from AirBERT [5], following the standard REVERIE generative navigation setting. This backbone supports cross-modal reasoning via co-attention and is well-suited for object-centric grounding.
- **Metrics:** We follow the official REVERIE [15] evaluation protocol and report *Success Rate (SR)*, *Success weighted by Path Length (SPL)*, *Remote Grounding Success (RGS)*, and *Remote Grounding Success weighted by Path Length (RGSPL)*. SR/SPL evaluate navigation success and efficiency, where a trajectory is counted as successful if the agent stops at a viewpoint where the target object is visible; RGS/RGSPL measure object grounding accuracy at the stopping viewpoint. We additionally report *Oracle Success Rate (OSR)*, which counts an episode as successful if any visited viewpoint satisfies the success condition.

CVDN Dataset. CVDN [21] extends the VLN task to a dialog-based navigation setting, where agents must interpret multi-turn dialog instructions to reach the

Table 2: Results on **REVERIE** (all metrics in %, higher is better). Client-side storage reports the maximum number of parameters stored on each client.

Method	SR ↑	SPL ↑	OSR ↑	RGSPL ↑	RGS ↑	Client Storage (Params)
Centralized	41.78	36.33	**46.65**	29.23	32.87	–
FedVLN	29.83	25.08	34.17	19.81	22.95	235.18M (100%)
pFedNavi	–	–	–	–	–	470.10M (199.9%)
per-Fedavg	12.51	11.33	13.76	4.71	5.10	235.18M (100%)
FedPerfix	37.22	31.40	40.67	26.24	31.18	253.92M (107.9%)
`FedNASP`	**43.23**	**36.62**	<u>45.88</u>	**30.70**	**35.22**	256.86M (109.2%)

goal location. Compared with single-instruction VLN datasets, CVDN exhibits stronger linguistic variability and longer contextual dependencies, which further amplifies non-IID instruction styles across clients in federated settings. More implementation details are in the supplementary.

- **Model:** We adopt a seq2seq-style VLN agent as the backbone following the standard CVDN navigation setting [21]. The model encodes dialog history using an LSTM-based language encoder and grounds navigation actions through attention over panoramic visual features.
- **Metrics:** We follow the standard CVDN evaluation protocol and report *Success Rate (SR)*, *Success weighted by Path Length (SPL)*, *Navigation Error (NE)*, *Oracle Success Rate (OSR)*, *Oracle Path Success Rate (OPSR)*, and *Dist-to-End Reduction (DER)*. SR, SPL, and OSR follow the same definitions as in REVERIE. NE measures the shortest-path distance between the agent's stopping location and the goal viewpoint. DER measures the reduction in distance to the goal achieved by the trajectory compared with the starting position.

Baselines: We compare our method with both centralized and federated learning approaches. For federated learning, we evaluate **FedVLN** [27], the first work that applies FedAvg [11] to the VLN setting. We further evaluate representative pFL baselines on REVERIE and CVDN: Per-FedAvg [2] as a classic method, FedPerfix [18] on REVERIE due to its similar adapter-based design, and FedCP [25] on CVDN because FedPerfix does not support non-transformer Seq2Seq settings.

5.1 The Effectiveness of `FedNASP`

REVERIE The results in Tab. 2 show that `FedNASP` consistently outperforms both federated and personalized baselines on REVERIE. Compared with FedVLN, `FedNASP` improves SR/SPL by 45%/46% and RGSPL/RGS by 55%/53%, demonstrating stronger navigation and object grounding under severe client heterogeneity.

To better understand where adaptive step-wise personalization is most beneficial, we further analyze performance on scan subsets selected by different heterogeneity factors (see supplementary §1.2 for factor definitions and subset construction). As Tab. 3 shows, `FedNASP` consistently improves both SR and

Table 3: Subset analysis on REVERIE. We report average SR and RGS (%) on four representative top-10 scan subsets selected by scan-level heterogeneity factors. Δ denotes the absolute improvement of FedNASP over FedAvg.

Subset	SR ↑			RGS ↑		
	FedAvg	FedNASP	Δ	**FedAvg**	FedNASP	Δ
Path mean	13.86	28.34	+14.48	11.61	22.07	+10.46
Instruction	22.61	33.33	+10.72	14.44	25.48	+11.04
High-branch ratio	14.88	32.77	+17.89	12.61	27.92	+15.31
Density	44.92	58.68	+13.76	37.51	52.20	+14.69

Table 4: Results on **CVDN** (SR, SPL, OSR, and OPSR in %). Higher is better except for NE.

Method	SR ↑	SPL ↑	NE ↓	OSR ↑	OPSR ↑	DER ↑	Len
Centralized	**43.56**	**33.13**	6.11	**62.85**	**76.20**	**5.63**	16.27
FedVLN	21.19	8.24	7.61	42.80	53.76	3.85	22.84
pFedNavi	19.61	13.62	9.71	45.54	55.87	2.58	14.02
per-Fedavg	22.16	10.56	7.58	43.42	53.19	4.27	28.67
FedCP	31.82	14.49	6.42	49.56	62.10	5.32	21.32
FedNASP	39.02	23.14	**5.98**	57.09	71.40	5.62	21.06

RGS across all subsets. The largest gains appear on high-branching and dense-graph scans, where agents must repeatedly resolve local ambiguities among multiple plausible actions. We also observe clear improvements on long-horizon and instruction-driven subsets, suggesting that adaptive step-wise personalization remains effective when navigation depends on accumulated history and instruction variability.

Communication and complexity analysis. Compared with FedVLN, FedNASP introduces lightweight client-specific components (SPI, SPM, and a critic), increasing the maximum client-side storage from 235.18M to 256.86M parameters (about +9.2%) in Tab. 2. Importantly, the communication pattern remains unchanged: clients upload only the shared backbone updates for FedAvg aggregation, while personalized components are kept local and never transmitted, avoiding additional communication overhead. In terms of runtime, FedNASP incurs negligible computational overhead: one federated round takes 14.37 minutes versus 14.12 minutes for FedVLN (about +1.8%), indicating that step-wise modulation and prefix generation add limited extra computation while yielding substantial performance gains.

CVDN Tab. 4 reports results on the CVDN dataset. Compared with the federated baseline FedVLN [27], FedNASP yields substantially better navigation performance across all metrics. In particular, FedNASP nearly doubles SR (+84%) and more than triples SPL (+180%), indicating that step-wise personalization improves both task completion and path efficiency. Importantly, FedNASP also consistently outperforms the pFL baseline pFedNavi, with large gains in SR/SPL

Table 5: Ablation on REVERIE and CVDN. All metrics are in % except for DER. Higher is better.

	REVERIE			CVDN		
Variant	**SR ↑**	**OSR ↑**	**RGS ↑**	**SR ↑**	**SPL ↑**	**DER ↑**
w/o SPI	29.83	34.17	22.95	21.19	8.24	3.85
w/o SPM	39.96	41.20	30.91	32.40	16.80	4.23
`FedNASP`	**43.23**	**45.88**	**35.22**	**39.02**	**23.14**	**5.62**

and substantially lower NE, demonstrating that our step-wise, structure-aware personalization is more effective for dialog-based, long-horizon navigation than using a fixed or coarse personalization strategy. In addition, `FedNASP` achieves DER comparable to centralized training while maintaining a shorter average trajectory length than FedVLN, indicating more effective progress toward the goal without excessive wandering. Although centralized training remains strongest in SR/SPL, `FedNASP` substantially closes the gap to centralized performance while preserving the federated setting, supporting the effectiveness of step-wise personalization under multi-round, heterogeneous CVDN training.

5.2 Ablation Study

We analyze the contribution of two key components in FedNASP: SPM and the adapter-based SPI. As summarized in Tab. 5, removing the SPI causes a pronounced drop in both navigation success and object grounding, indicating that SPI is essential for effective personalization. Disabling SPM while keeping the prefix still degrades performance, suggesting that a fixed personalization strength is suboptimal for long-horizon navigation. Overall, the `FedNASP` performs best, showing that step-wise modulation and structure-aware prefix injection are complementary for federated VLN. Additional analyses are provided in the supplementary material: 1) We analyze different SPM input configurations in §2.1, showing that both trajectory history and current observations are necessary for effective step-wise personalization; 2) In §2.2, we show the stable performance of `FedNASP` across different federated client sampling fractions; (3) To confirm the role of SPM in learning step-wise personalization strategies, we visualize SPM dynamics across communication rounds and within a single navigation episode in §2.3. The varying curves show that SPM indeed learns step-wise personalization rather than collapsing to a constant value; (4) To make a fair comparison with pFedNavi, additional experiments are conducted on the shared R2R benchmark in §2.4.

6 Conclusion

This paper investigates personalized federated learning for VLN under realistic deployment settings, where each client represents a distinct indoor environment

and data are highly non-IID. To overcome the limits of a single global policy and static personalization, we propose FedNASP, a step-wise personalized FL framework that dynamically adjusts personalization strength along the navigation trajectory. With a lightweight SPM and structure-aware adapter-based personalized prefix injection, FedNASP enables fine-grained adaptation to time-varying decision contexts while preserving globally transferable knowledge in a shared backbone. Experiments on REVERIE, CVDN, and R2R show that FedNASP consistently improves over strong federated VLN and personalized FL baselines. Comprehensive ablations and rollout visualizations further demonstrate the effectiveness of adaptive step-wise personalization. Overall, adaptive step-wise personalization offers an effective and practical path toward scalable, privacy-preserving federated VLN in diverse real-world environments.

Acknowledgments

The work of Qingqian Yang and Hao Wang was supported in part by the United States National Science Foundation (NSF) under grants 2523997, 2315612, and 2332638 and by the NVIDIA Academic Grant Program using RTX GPUs. This work of Tao Song was supported in part by the National Natural Science Foundation of China (NO. 62472284), Openmind (Wuhu) Intelligent Robot Co., Ltd., and Shanghai Key Laboratory of Scalable Computing and Systems. The work of Miao Pan was supported in part by the US National Science Foundation under grants CNS-2107057, CNS-2318664, CSR-2403249, and CNS-2431596. Hao Wang is the corresponding author.

References

1. Anderson, P., Wu, Q., Teney, D., Bruce, J., Johnson, M., Sünderhauf, N., Reid, I.D., Gould, S., van den Hengel, A.: Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In: CVPR (2018)
2. Arivazhagan, M.G., Aggarwal, V., Singh, A.K., Choudhary, S.: Federated learning with personalization layers. arXiv preprint arXiv:1912.00818 (2019)
3. Chang, A., Dai, A., Funkhouser, T., Halber, M., Niessner, M., Savva, M., Song, S., Zeng, A., Zhang, Y.: Matterport3d: Learning from rgb-d data in indoor environments. arXiv preprint arXiv:1709.06158 (2017)
4. Gao, X., Gao, Q., Gong, R., Lin, K., Thattai, G., Sukhatme, G.S.: Dialfred: Dialogue-enabled agents for embodied instruction following. IEEE Robotics and Automation Letters **7**(4), 10049–10056 (2022)
5. Guhur, P.L., Tapaswi, M., Chen, S., Laptev, I., Schmid, C.: Airbert: In-domain pretraining for vision-and-language navigation. In: ICCV (2021)
6. He, K., Si, C., Lu, Z., Huang, Y., Wang, L., Wang, X.: Frequency-enhanced data augmentation for vision-and-language navigation. In: NeurIPS (2026)
7. Irshad, M.Z., Ma, C.Y., Kira, Z.: Hierarchical cross-modal agent for robotics vision-and-language navigation. In: ICRA (2021)
8. Krantz, J., Anderson, P., Shrivastava, A., Batra, D., Parikh, D., Lee, S.: Beyond the nav-graph: Vision-and-language navigation in continuous environments. In: ECCV (2020)

9. Ku, A., Anderson, P., Patel, R., Ie, E., Baldridge, J.: Room-across-room: Multilingual vision-and-language navigation with dense spatiotemporal grounding. In: EMNLP (2020)
10. Li, T., Hu, S., Beirami, A., Smith, V.: Ditto: Fair and robust federated learning through personalization. In: ICML (2021)
11. 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 (2017)
12. Miao, C., Chang, T., Wu, M., Xu, H., Li, C., Li, M., Wang, X.: Fedvla: Federated vision-language-action learning with dual gating mixture-of-experts for robotic manipulation. In: ICCV (2025)
13. Nguyen, K., Daumé III, H.: Help, anna! visual navigation with natural multimodal assistance via retrospective curiosity-encouraging imitation learning. In: EMNLP-IJCNLP. pp. 684–695 (2019)
14. Padmakumar, A., Thomason, J., Shrivastava, A., Lange, P., Narayan-Chen, A., Gella, S., Piramuthu, R., Tur, G., Hakkani-Tur, D.: Teach: Task-driven embodied agents that chat. In: AAAI (2022)
15. Qi, Y., Wu, Q., Anderson, P., Wang, X., Wang, W.Y., Shen, C., Hengel, A.v.d.: Reverie: Remote embodied visual referring expression in real indoor environments. In: CVPR (2020)
16. Ramakrishnan, S.K., Gokaslan, A., Wijmans, E., Maksymets, O., Clegg, A., Turner, J., Undersander, E., Galuba, W., Westbury, A., Chang, A.X., et al.: Habitat-matterport 3d dataset (hm3d): 1000 large-scale 3d environments for embodied ai. arXiv preprint arXiv:2109.08238 (2021)
17. Shridhar, M., Thomason, J., Gordon, D., Bisk, Y., Han, W., Mottaghi, R., Zettlemoyer, L., Fox, D.: Alfred: A benchmark for interpreting grounded instructions for everyday tasks. In: CVPR (2020)
18. Sun, G., Mendieta, M., Luo, J., Wu, S., Chen, C.: Fedperfix: Towards partial model personalization of vision transformers in federated learning. In: ICCV (2023)
19. T Dinh, C., Tran, N., Nguyen, J.: Personalized federated learning with moreau envelopes. In: NeurIPS (2020)
20. Tan, H., Yu, L., Bansal, M.: Learning to navigate unseen environments: Back translation with environmental dropout. In: NAACL-HLT (2019)
21. Thomason, J., Murray, M., Cakmak, M., Zettlemoyer, L.: Vision-and-dialog navigation. In: CoRL (2020)
22. Wang, L., Xia, X., Zhao, H., Wang, H., Wang, T., Chen, Y., Liu, C., Chen, Q., Pang, J.: Rethinking the embodied gap in vision-and-language navigation: A holistic study of physical and visual disparities. In: CVPR (2025)
23. Yang, Q., Wang, H., Zhang, S.Q., Li, J., Hua, Y., Pan, M., Song, T., Qi, Z., Guan, H.: pfednavi: Structure-aware personalized federated vision-language navigation for embodied ai. arXiv preprint arXiv:2602.14401 (2026)
24. Yin, H., Xu, X., Zhao, L., Wang, Z., Zhou, J., Lu, J.: Unigoal: Towards universal zero-shot goal-oriented navigation. In: CVPR (2025)
25. Zhang, J., Hua, Y., Wang, H., Song, T., Xue, Z., Ma, R., Guan, H.: Fedcp: Separating feature information for personalized federated learning via conditional policy. In: Proceedings of the 29th ACM SIGKDD conference on knowledge discovery and data mining. pp. 3249–3261 (2023)
26. Zhou, G., Hong, Y., Wang, Z., Wang, X.E., Wu, Q.: Navgpt-2: Unleashing navigational reasoning capability for large vision-language models. In: ECCV (2024)
27. Zhou, K., Wang, X.E.: FedVLN: Privacy-preserving federated vision-and-language navigation. In: ECCV (2022)

28. Zhu, F., Liang, X., Zhu, Y., Yu, Q., Chang, X., Liang, X.: Soon: Scenario oriented object navigation with graph-based exploration. In: CVPR (2021)