Commun Transp Res

Research Article

A federated meta-learning approach for interpretable, privacy-preserving, and customizable behavior analysis

Linlin You[1,2], Kunxu Chen[1,2], Baichuan Mo[3,4,✉], Jiemin Xie[1,2], Juanjuan Zhao[5,✉], Jinhua Zhao[6]

Cite this article: You L L, Chen K X, Mo B C, et al. *Commun Transp Res* 2026, **6**(1): 9640014. https://doi.org/10.26599/COMMTR.2026.9640014

ABSTRACT: Travel behavior analysis provides critical insights to enhance the intelligence of transportation systems, enabling more accurate and efficient management of mobility services. However, it requires centralizing user-sensitive data which may violate regulations and laws about data security. Even though various solutions have been proposed to train deep neural networks (DNNs) via federated learning, it still faces three critical challenges in ensuring the interpretability of DNNs to unfold the black-box, bridging isolated data to train adaptive model, and harnessing the heterogeneity among users to support personalized analysis. To tackle these challenges, this study proposes an interpretable, privacy-preserving and customizable approach to support travel behavior analysis based on federated meta-learning, called IPC-FM. Specifically, it, first, introduces an artificial neural network empowered with three kinds of utilities associated with discrete choice models to provide interpretable results. Second, it integrates federated meta-learning to train a globally meta-model via the knowledge among clients in a collaborative and privacy-preserving manner. Finally, it enables rapid model localization to support personalized analysis. Based on standard datasets, IPC-FM is evaluated against state-of-the-art methods. The results show that IPC-FM can collaborate clients with isolated and heterogeneous data to train a robust, customizable and interpretable model for travel behavior analysis.

KEYWORDS: federated learning (FL); artificial neural network; federated meta-learning; travel behavior analysis; discrete choice model

1 Introduction

Behavior analysis plays a crucial role across various domains, including economics and transportation (McFadden, 1974; Train, 2009). In transportation, travel behavior analysis provides critical insights that help enhance the intelligence of transportation systems (You et al., 2024). Its primary aim is to predict individual mobility demand and subsequently develop effective management strategies (Park and Reisinger, 2010) to improve travel quality and alleviate traffic congestion (Li et al., 2020). Conventionally, discrete choice models (DCMs) have been widely adopted to predict travel choices and analyze underlying decision-making processes (Train, 2009). However, in recent years, as deep neural networks (DNNs) have demonstrated exceptional performance in tasks such as computer vision and natural language processing, researchers have been attempting to explore DNNs as alternatives to traditional DCMs (Wang et al., 2020b) to capture complex relationships in user data for more accurate analysis results.

Despite the evident potential of DNNs in travel behavior analysis, several challenges, as illustrated in Fig. 1, impede their effective application. First, DNNs, in general, work as a "black box" that prioritizes prediction performance at the expense of interpretability (Elharoun et al., 2023; Wang et al., 2024), limiting insights into the factors influencing user decisions. Second, in conventional methods, user sensitive data, such as economic status and social behavior attributes, need to be collected and processed in a centralized way. This introduces a single point of failure, where sensitive data are aggregated in one place, making them more attractive and vulnerable to large-scale breaches. Accordingly, data security regulations (Sethu, 2020) restrict the sharing of data. Meanwhile, the continuous enhancement of public awareness regarding data privacy makes individuals increasingly reluctant to share their personal data. This approach limits the exposure risk, ensuring that potential leakage is confined to a single user rather than compromising the entire system. However, this resistance undermines previous app-based mobility analysis solutions, such as the future mobility sensing system platform (You et al., 2018) and MOVE-ME application (Cunha and Galvão, 2014). Consequently, data sensed and stored at personal devices tend to form data silos, making centralized model training infeasible (Yang et al., 2019). Finally, the efficacy of behavior analysis is fundamentally challenged by user heterogeneity. This concept refers to the inherent diversity in individual preferences and decision-making processes. In the

[1] School of Intelligent Systems Engineering, Sun Yat-sen University, Shenzhen 518107, China. [2] Guangdong Provincial Key Laboratory of Intelligent Transportation Systems, Sun Yat-sen University, Guangzhou 510275, China. [3] Department of Civil Engineering, Tsinghua University, Beijing 100084, China. [4] Department of Civil and Environmental Engineering, Massachusetts Institute of Technology, Cambridge MA 02139, USA. [5] College of Resource Environment and Tourism, Capital Normal University, Beijing 100048, China. [6] Department of Urban Studies and Planning, Massachusetts Institute of Technology, Cambridge MA 02139, USA.

✉ Corresponding authors. E-mail: C. B. Mo, baichuan@mit.edu; J. J. Zhao, zhao@cnu.edu.cn

Received: October 11, 2025; Revised: November 29, 2025; Accepted: January 28, 2026

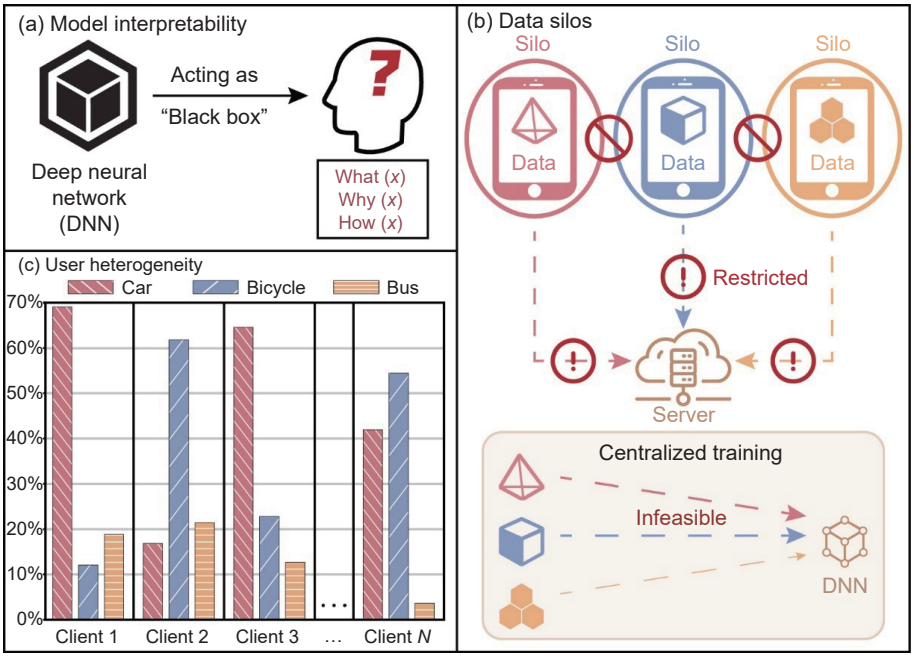


Fig. 1 Challenges to applying DNNs in travel behavior analysis.

context of travel behavior, it manifests as significant variations in the distribution of choices across different users (Jian et al., 2025; Liu et al., 2024). The mixed logit (MXL) model, a cornerstone of modern discrete choice analysis, provides a principled mathematical framework to capture this heterogeneity. Its core principle is to model the population not with a single set of average parameters but by assuming that individual-specific preference parameters follow a continuous distribution (e.g., a multivariate normal distribution characterized by a mean and a covariance matrix). However, while the MXL model effectively represents heterogeneity, its conventional centralized estimation paradigm requires pooling all user-sensitive data in one location, which is increasingly infeasible due to data privacy regulations and distributed data silos.

To address these challenges, this study introduces IPC-FM, an interpretable, privacy-preserving and customizable method based on federated meta-learning (FM), which integrates a threefold utility-engaged artificial neural network (T-ANN) and a three-step meta-model utilization procedure (T-MUP). First, as a privacy-preserving learning environment, the data at each user are too sparse for DNNs to capture key knowledge. T-ANN is designed to not only maintain the analysis process of DCMs but also incorporate dedicated embedding components to better extract hidden patterns from user data for more accurate analysis. Second, for privacy preservation, the framework is architected with a dual-layer protection strategy. It adopts federated learning (FL) as its foundational data-level defense, ensuring that all user-sensitive data remain on local devices and are never centralized, thereby eliminating the risk of large-scale data breaches. Building upon this, T-MUP introduces a complementary model-level protection. By structuring the

local training around support and query sets, the model updates shared with the server are inherently derived from an abstracted learning process, which inherently obfuscates the direct mapping to raw data and fortifies the framework against potential privacy inference attacks. Furthermore, to achieve customizability, the T-MUP procedure addresses the intrinsic user heterogeneity that FL alone cannot resolve. By integrating meta-learning, T-MUP enables systematic knowledge consolidation across clients to train a globally shareable meta-model. This meta-model can then be rapidly adapted with minimal local data, supporting fast, personalized model adaptation to heterogeneous user contexts and thus bridging the gap between global generalization and local personalization. In general, the main contributions of this study can be summarized as follows:

- A threefold utility-engaged ANN (T-ANN) is designed to extract both linear and nonlinear relationships between different features encoded in user sensitive data. The model can not only retain interpretability similar to conventional DCMs but also enhance the prediction capability for more accurate results;
- The model training is empowered by a three-step meta-model utilization procedure (T-MUP) based on federated meta-learning, which not only protects user private data but also addresses user heterogeneity. Initially, it collaborates clients to train a global meta-model in a privacy-preserving manner. Subsequently, each user can swiftly localize the meta-model to enable more personalized behavior analysis;
- The performance of IPC-FM is evaluated on standard datasets. Specifically, the proposed method outperforms state-of-the-art (SOTA) methods (i.e., MNL, ASU-DNN (Wang et al., 2020a), E-MNL (Arkoudi et al., 2023), and L-MNL (Sifringer et al., 2020)) with average improvements of 3% in accuracy, 10% in test loss, 11% in F1 score, and 7% in kappa score. Moreover, the interpretability of IPC-FM is also ensured to support the analysis of reasons behind user choice behaviors.

The rest of the paper is organized as follows. First, Section 2 summarizes related work. Second, IPC-FM is presented and evaluated in Sections 3 and 4, respectively. Finally, Section 5 concludes the work and discusses the future.

2 Related work

This section summarizes the related research on DNNs supporting travel behavior analysis and then discusses federated learning together with meta-learning.

2.1 DNNs for behavior analysis

In recent years, DNNs have been increasingly applied to support various transportation tasks, such as traffic flow forecasting (Ma et

al., 2021) and travel behavior analysis (Martín-Baos et al., 2023; Moreau et al., 2022). Compared to traditional methods, such as DCMs, DNNs can significantly enhance their prediction accuracy by employing more complex model structures (Zhao et al., 2020). Nevertheless, DNNs are often considered "black-box" models, which lack transparency and interpretability (Salih et al., 2024).

From the perspective of travel behavior analysis, model interpretability is crucial, as it helps analysts and managers uncover the reasons behind user behaviors (Park and Reisinger, 2010). Conventionally, statistical models have been widely studied to quantitatively reveal the importance of factors affecting user choice behavior (McFadden, 1974). Motivated by that, scholars have sought to design DNNs that can represent DCMs to achieve high prediction ability with interpretability remaining. Recently, Wang et al. (2021) focused on extracting economic indicators, such as marginal effects, from DNNs to explore the theoretical transition from DCMs to DNNs and analyze the trade-off between predictability and interpretability. After that, Wang et al. (2020a) and Wong and Farooq (2021) attempted to design dedicated DNNs to represent DCMs that can retain interpretability without sacrificing prediction performance. Several notable works include Learning Multinomial Logit (L-MNL) (Sifringer et al., 2020) and its enhanced version with Embedding (E-MNL) (Arkoudi et al., 2023), which encode individual variables into continuous feature vectors that can be directly associated with related choices. By simulating the analysis process of DCMs, these models can maintain straightforward interpretability. Beyond these efforts, recent research has also introduced post hoc explanation techniques such as Shapley value-based methods, which attribute the contribution of each feature to mode predictions in a manner consistent with cooperative game theory (Sundararajan and Najmi, 2020). Moreover, causal inference integration has emerged as another promising approach, where structural causal models and counterfactual reasoning are combined with machine learning to provide deeper behavioral insights and policy-relevant interpretability in travel mode choice and related transport applications (Chauhan et al., 2023; Kamal and Farooq, 2024). However, current models are generally designed to handle linear features and are incapable of processing nonlinear relationships hidden in heterogeneous user data for more accurate analysis.

2.2 FL and meta-learning

As a novel distributed learning paradigm, FL (McMahan et al., 2017; Yang et al., 2019) is attracting increasing attention from both academia and industry. When training the model, FL eliminates the need to gather and process user data in a central server. Instead, each client can train an individual model based on its local data and then share the model with the server to update the global model (Jian et al., 2025; You et al., 2022). In general, FL can significantly mitigate the risk of sensitive data exposure by enabling individuals to collaboratively and privately exchange local knowledge (Lin et al., 2025). Despite its advantages in ensuring user privacy, FL still faces challenges arising from data heterogeneity (Li et al., 2019). This heterogeneity, caused by differences in data distribution among clients, can negatively impact model performance across diverse tasks. Several studies have explored and proposed related solutions to address the heterogeneity within distributed data (Acar et al., 2021; Gao et al., 2022), but few are designed to support personalized travel behavior analysis.

To enable the optimization of initial model parameters for rapid model adaptation (Hospedales et al., 2022), meta-learning has been discussed and utilized to train models for classification, regression, and reinforcement learning (Finn et al., 2017). Since conventional meta-learning algorithms still need to process data in a centralized way, researchers combine FL with meta-learning, named FM, to bridge data silos caused by privacy protection and train a globally shareable meta-model (Fallah et al., 2020; Li et al., 2022). Currently, FM-empowered methods have been studied to support several transportation tasks, e.g., predicting parking occupancy (Qu et al., 2022) and detecting driver distraction (Liu et al., 2024), which show the merit of FM in providing personalized solutions for each user. However, how to utilize FM to train an interpretable model for personalized travel behavior analysis is still an open question.

3 Methodology

As shown in Fig. 2, IPC-FM consists of (1) a threefold utility-engaged ANN (T-ANN) working as the backbone to encode key knowledge extracted from heterogeneous data and (2) a three-step meta-model utilization procedure (T-MUP) designed for training and localizing a shareable meta-model.

3.1 Preliminaries

To establish the necessary theoretical foundation for the proposed framework, this section introduces three key components: DCMs, FL, and meta-learning. DCMs serve as a mathematical method for modeling individual decision-making behavior among a finite set of alternatives. FL provides a decentralized learning paradigm that ensures data privacy by keeping raw data on local devices. Meta-learning equips models with the ability to rapidly adapt to new tasks with limited data, which is essential in personalized and heterogeneous decision-making scenarios. These components collectively form the basis for the integrated approach proposed in this study.

3.1.1 Discrete choice model

DCMs can be derived from random utility maximization theory (McFadden, 1974), which assumes that all individuals maximize their utilities to make their choices. Let $U_{n,j}$ be the utility of individual n for travel alternative j. $U_{n,j}$ is assumed to be decomposed into two parts:

$$U_{n,j} = V_{n,j} + \varepsilon_n, \ \forall j \ \in C_n, n = 1, \ldots, N \tag{1}$$

where $V_{n,j}$ stands for the systematic utility and ε is a random component. $C_n = \{1,2, \ldots, J\}$ is the alternative set of individual n. One of the most widely used DCMs is the multinomial logit (MNL) model, which assumes that ε_n follows an independent and identically distributed Type I extreme value distribution. In this case, the probability of individual n choosing alternative j can be expressed as

$$P_{n,j} = \mathbb{P}\left[U_{n,j} \geq U_{n,j'}, \forall j' \in C_n\right] = \frac{\exp\left(V_{n,j}\right)}{\sum_{j'=1}^{J} \exp\left(V_{n,j'}\right)} \tag{2}$$

In a typical MNL model, $V_{n,j}$ is usually quantified by a linear formulation, which cannot capture complex nonlinear relationships between features and decisions.

3.1.2 FL for privacy protection

FL is a privacy-preserving distributed training framework that enables multiple clients to collaboratively train a shared global model without sharing their raw data (McMahan et al., 2017). The client is the device that requests or receives a service, and the

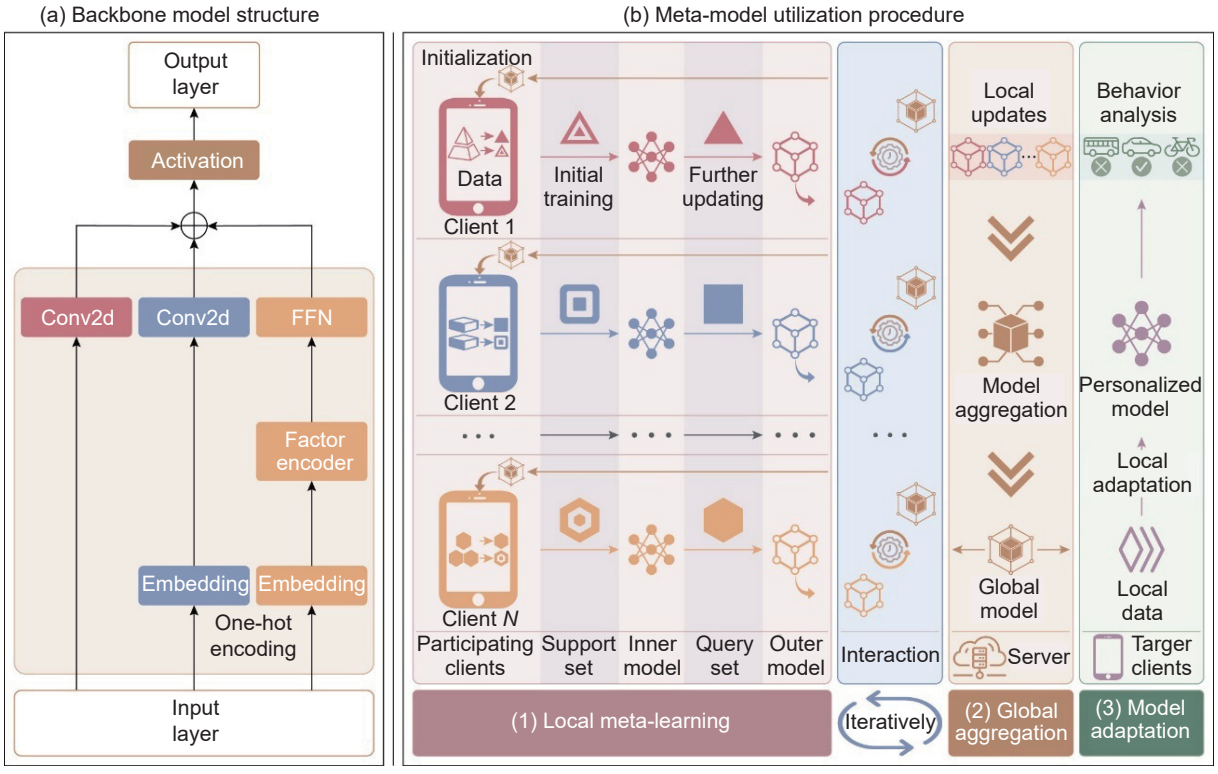


Fig. 2 Overall architecture of IPC-FM: (a) backbone model structure, where FFN stands for the feed-forward network; (b) meta-model utilization procedure, which includes local meta-learning, global aggregation and model adaptation.

training process is typically orchestrated in a server-client architecture, where each client performs local model training and the central server is responsible for aggregating model parameters.

In FL, each client $n \in \{1, 2, \ldots, N\}$ locally trains a model using its own dataset $\mathcal{D}_n$ and periodically shares the model parameters with a central server. Let θ_i^{global} denote the global model parameters at communication round i. At the beginning of each round, the server broadcasts θ_i^{global} to a selected subset of clients as their initialization model. Each participating client n uses its local dataset to update the model as defined in Eq. (3):

$$\theta_{i,n,\tau+1} \leftarrow \theta_{i,n,\tau} - \eta \nabla_{\theta_{i,n,\tau}} \mathcal{L}_{\mathcal{D}_n}(\theta_{i,n,\tau}), \ \forall \tau = 0, \ldots, R-1 \quad (3)$$

where η is the learning rate, $\mathcal{L}_{\mathcal{D}_n}(\cdot)$ denotes the local loss function of client n, and τ and R represent the current and the total number of local training iterations, respectively.

After completing local training, each client sends the updated model $\theta_{n,i,R}$ to the server. Then, the server aggregates the received models from all the clients by using a weighted strategy to update the global model parameters $\theta_{i+1}^{\text{global}}$, as defined in Eq. (4):

$$\theta_{i+1}^{\text{global}} = \sum_{n=1}^{N} \left(\gamma_n \times \theta_{i,n}\right), \forall i = 0, \ldots, I-1 \quad (4)$$

where γ_n represents the aggregation weight of client n, and I is the total number of global communication rounds.

Once the global model $\theta_{i+1}^{\text{global}}$ is updated, it is broadcast to all the clients to initiate the next round of local training. This process iterates until the$I$ communication round is reached. Since only model parameters are transmitted between the server and clients during the training process, rather than raw user data, it enables FL to reduce privacy risks and mitigate regulatory concerns.

3.1.3 *Meta-learning for rapid adaptation*

Meta-learning, or "learning to learn" is a learning paradigm aimed at training models that can rapidly adapt to new tasks using only a small amount of data (You et al., 2025). In contrast to conventional training methods that focus on optimizing model performance on a single task, as shown in Fig. 3, meta-learning involves a dual-level training structure: an inner loop for task-specific adaptation and an outer loop for learning transferable knowledge across tasks.

Formally, let $\mathcal{D} = \{(\mathcal{D}_s, \mathcal{D}_q)\}$ represent a task, where $\mathcal{D}_s$ is the support set and $\mathcal{D}_q$ is the query set. In general, the task within meta-learning can be categorized into training and testing tasks. Based on a set of training tasks, the meta-model can be trained. After that, testing tasks can be used to validate the adaptivity of the meta-model, as well as its task-specific performance, where the support and query sets are used for model adaptation and performance evaluation, respectively. In the context of travel behavior analysis, a "task" corresponds to predicting the travel mode choice of a particular user. For instance, the support set $\mathcal{D}_s$ contains a few observed user trips (e.g., commuting to work or school) for adapting the model to the preferences of this user, while the query set $\mathcal{D}_q$ holds out separate trips for assessing the generalization of the model to future choices. This setup mimics real-world scenarios where only limited information is available for a new traveler, and the model must still produce reliable predictions. Given an initial model parameter θ, the inner-loop

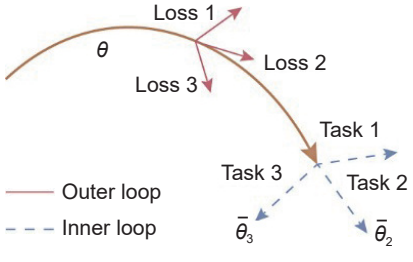


Fig. 3 Updating process of model parameters under meta-learning.

adaptation updates the model on the support set as defined in Eq. (5):

$$\overline{\theta} = \theta - \alpha \nabla_{\theta} \mathcal{L}_{\mathcal{D}_s}(\theta) \tag{5}$$

where α is the inner-loop learning rate.

The updated parameter $\overline{\theta}$ is then evaluated on the query set $\mathcal{D}_q$ to fulfill the meta-objective. In general, the outer loop summarizes feedback from multiple data samples, each of which represents a specific learning task, to update the meta-parameter θ as defined in Eq. (6):

$$\theta \leftarrow \theta - \beta \nabla_{\theta} \mathcal{L}_{\mathcal{D}_q}\left(\overline{\theta}\right) \tag{6}$$

where β is the outer-loop learning rate.

This optimization enables the model to acquire an initial parameter θ that is sensitive to task-specific gradients, thereby allowing for fast adaptation to new, unseen tasks after only a few gradient steps. In the transportation setting, this means that when the model encounters a new user whose behavior has not been seen before, it can still rapidly adjust to provide accurate predictions that are suitable for travel preferences. This capability is particularly valuable in personalized or heterogeneous environments.

3.2 Learning problem definition

Considering the importance of data security and privacy, this study strictly prohibits the sharing and exchange of user sensitive data. Under this setting, the objective of this study is to train an interpretable model in a privacy-preserving manner for user behavior analysis. Consequently, the model should satisfy the following three restrictions simultaneously:

- **Interpretability**: The model needs to be interpretable to reveal the reason behind a user travel choice;
- **Privacy protection**: Each client protects its private data by only exchanging model parameters updated locally with the server, rather than their raw data;
- **Adaptability**: The model needs to be adaptable and can be rapidly customized to fit the local contexts of users in expressing individual and heterogeneous preferences.

To ensure interpretability, we design a T-ANN that follows the same structure as the MNL model with the exception of assuming $V_{n,j} = f_j(\cdot|\theta)$, where $f_j(\cdot|\theta)$ is a threefold ANN architecture that will be elaborated in Section 3.3. It enables the model to capture complex nonlinear relationships in decision making. To protect private data and address user heterogeneity, we design T-MUP based on FM to collaboratively train and apply the model in three steps, namely, local meta-learning, global model aggregation and model adaptation. Specifically, for each client n, it initializes the global model as its local model and then starts local meta-learning based on local support set $\mathcal{D}_{n,s}$ and query set $\mathcal{D}_{n,q}$. The support set contains a few examples of each choice alternative in the client dataset, allowing the local model to quickly capture behavior patterns. The query set is the client dataset in addition to the support set, helping the model learn how to quickly adapt to different travel choices and produce a model that generalizes better. After local models are trained and uploaded to the server, they are aggregated to update the global model. Finally, when the global model is trained, it is dispatched to the target clients, who will adapt the model based on their local data to perform personalized travel behavior analysis. In this process, cross-entropy is often used as the learning loss to optimize the parameters of the model during the backpropagation process. Thus, the learning loss function is defined as Eq. (7):

$$\min_{\theta} \sum_{n}^{N} \mathcal{L}_{\mathcal{D}_{n,q}}(\theta) = \min_{\theta} \sum_{n}^{N} \sum_{d}^{\mathcal{D}_{n,q}} \sum_{j}^{J} -y_{n,d,j} \cdot \log\left(P_{n,d,j}\right) \tag{7}$$

where $y_{n,d,j}$ is the label of choice alternative j for individual n in sample d, and $P_{n,d,j}$ is the predicted probability of individual n choosing alternative j in sample$d$.

3.3 T-ANN: Backbone model of IPC-FM

As shown in Fig. 4, T-ANN, the threefold utility-engaged ANN, is designed based on multinomial logit models (MNLs) and deep neural networks (DNNs) to maintain and improve the straightforward interpretability and predictability of the model, respectively. In general, T-ANN consists of four dedicated modules, namely, (1) a choice property module, which measures the utility related to choice alternatives; (2) a personal attribute module, which calculates the impact of personal attributes (e.g., age, gender); (3) a random coefficient module, which applies a transformer encoder and FFN (feed-forward network) to capture the randomness related to the heterogeneous travel choices among individuals; and (4) a concatenation and activation module, which integrates the outputs of the previous three models to generate the choice probability.

Within T-ANN, $V_{n,j}$ can be reformed as defined in Eq. (8):

$$V_{n,j} = f_j^X(X_n) + f_j^Q(Q_n) + f_j^T(T_n) \tag{8}$$

where $X_n \in \mathbb{R}^{K \times J}$ represents the choice properties of individual n when selecting choice alternative j; $Q_n \in \mathbb{R}^M$ denotes the personal attributes of individual n; $T_n \in \mathbb{R}^G$ stands for the latent factors of individual n that encode nonlinear relationships within user data; and K, M, and G are the lengths of related feature vectors. Specifically, $f_j^X(X_n)$ denotes the linear utility derived from choice-specific attributes (e.g., travel time and cost), $f_j^Q(Q_n)$ represents the linear utility associated with personal attributes (e.g., age and income) by embedding, and $f_j^T(T_n)$ captures the nonlinear utility from latent factors through self-attention and FFN. Finally, f^X, f^Q and f^T are the functions to be trained by the choice property, personal attribute, and random coefficient modules to calculate their corresponding utilities, respectively.

3.3.1 Choice property module

The function $f_j^X(X_n)$, as defined in Eq. (9), is relatively simply to calculate.

$$f_j^X\left(X_n|\boldsymbol{\theta}^X\right) = \left(\boldsymbol{\theta}^X\right)' \cdot \boldsymbol{x}_{n,j} \tag{9}$$

where $\boldsymbol{\theta}^X \in \mathbb{R}^K$ is the coefficient vector to be learned with respect to the choice properties.

3.3.2 Personal attribute module

Since it is difficult to establish a direct relationship between the personal attributes Q_n and the choice alternatives, we introduce an embedding layer $E^Q(\cdot)$ to match. To support the numerical calculation supported by $E^Q(\cdot)$, Q_n needs to be converted into a one-hot encoding form $Q_{n,\text{one-hot}} \in \mathbb{R}^{M \times S}$. Accordingly, $f_j^Q(Q_n)$ can be defined as Eq. (10):

$$f_j^Q(Q_n) = \left(\boldsymbol{\theta}^Q\right)' \cdot E_j^Q\left(Q_{n,\text{one-hot}}; \mathbf{W}_j^Q\right) \tag{10}$$

where $\boldsymbol{\theta}^Q \in \mathbb{R}^M$ is the coefficient vector to be learned with respect to the personal attributes, the embedding matrix $\mathbf{W}_j^Q \in \mathbb{R}^{S \times 1}$, and S represents the total number of personal attributes.

The embedding layer $E_j^Q\left(Q_{n,\text{one-hot}}; \mathbf{W}_j^Q\right) \in \mathbb{R}^{M \times 1}$ is computed

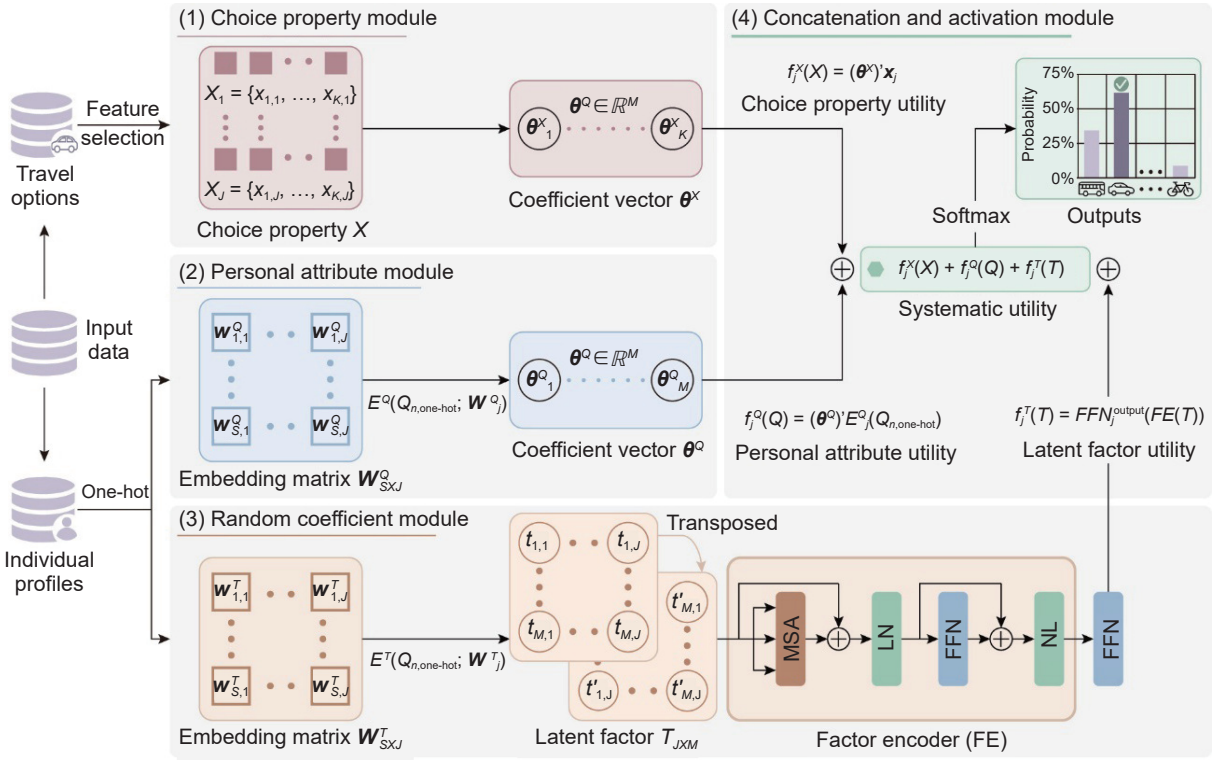


Fig. 4 Structure of the threefold utility engaged ANN, where MSA stands for the multihead self-attention, FFN stands for the feed-forward network, and LN stands for layer normalization.

as a linear projection of the one-hot input. The formulation of E_j^Q is defined as Eq. (11):

$$E_j^Q\left(Q_{n,\text{one-hot}};\mathbf{W}_j^Q\right)=Q_{n,\text{one-hot}}\cdot\mathbf{W}_j^Q \tag{11}$$

3.3.3 *Random coefficient module*

A factor encoder (FE) layer (Vaswani et al., 2017) is designed to explore nonlinear relationships hidden in user heterogeneous data as defined in Eq. (12):

$$T_{n,j}=E_j^T\left(Q_{n,\text{one-hot}};\mathbf{W}_j^T\right) \tag{12}$$

where $T_{n,j}\in\mathbb{R}^{M\times 1}$ is the extracted latent factor and E_j^T is an embedding layer with its embedding matrix $\mathbf{W}_j^T\in\mathbb{R}^{S\times 1}$.

To mine the correlation between different choice alternatives, we define $T_n=((T_{n,1})',\ldots(T_{n,J})')\in\mathbb{R}^{J\times M}$. Then, we obtain the feature matrix according to Eq. (13):

$$\begin{aligned}\overline{T}_{n,l}&=\text{LN}\left(\text{MSA}_l\left(T_{n,l-1}\right)+T_{n,l-1}\right),\forall l=0,\ldots,L\\ T_{n,l}&=\text{LN}\left(\text{FFN}_l^{\text{FE}}\left(\overline{T}_{n,l}\right)+\overline{T}_{n,l}\right),\forall l=0,\ldots,L\end{aligned} \tag{13}$$

where $T_{n,0}$ is equal to T_n, and $T_{n,L}\in\mathbb{R}^{J\times M}$, $\overline{T}_{n,l}$ and $T_{n,l}$ are the outputs of MSA (multihead self-attention) and FFN^{FE} (feed-forward network) layers after LN (layer normalization) in block l of individual n, respectively.

To fully exploit the information extracted from the FE, its output $T_{n,L}$ is further passed through an FFN layer. By default, the FFN adopts ReLU as its activation function. Accordingly, the utility component associated with the extracted features, denoted as $f^T(T_n)$, can be defined as shown in Eq. (14):

$$f^T(T_n)=\text{FFN}^{\text{output}}(T_{n,L})\in\mathbb{R}^{J\times 1} \tag{14}$$

Finally, we concatenate the three kinds of utilities to generate the systematic utility $V_{n,j}$ (as defined in Eq. (8)). This value is then transformed by a softmax activation function to generate $P_{n,j}$, which denotes the probability of individual n choosing alternative j, as given in Eq. (2).

3.4 T-MUP: Model utilization procedure of IPC-FM

IPC-FM integrates federated learning and meta-learning to first train an initial meta-model of T-ANN and then localizes it for personalized T-ANNs to support analysis in local contexts of individuals. During the training process, unlike centralized methods, IPC-FM needs clients to share model parameters updated locally instead of the raw data and allows clients to customize their model rapidly. where the parameters θ denote the set of all trainable parameters in the T-ANN, including the coefficient vector for choice attributes $\boldsymbol{\theta}^X$, the coefficient vector for personal attributes $\boldsymbol{\theta}^Q$, the embedding matrix $\mathbf{W}_j^Q$, and the model parameters associated with the random coefficient module. In general, to gain the above-described benefits, IPC-FM implements a training process with three steps, namely, (1) local meta-learning, (2) global aggregation, and (3) model adaptation.

3.4.1 *Local meta-learning*

In federated learning, each client typically possesses a limited amount of training data. In such data-scarce scenarios, standard training approaches are prone to overfitting and fail to effectively capture personalized behavioral patterns. To address this limitation, we adopt a meta-learning strategy that enables each client to perform rapid adaptation to its local data distribution using only a small number of samples.

When client n participates in the training process, it will initialize the learning according to the configuration (e.g., the model architecture, hyperparameters) dispatched by the server. Then, the client clones the received global model θ^{global} as its local model θ_n. At communication round i, the client trains the

Algorithm 1 Pseudocode of IPC-FM.

$\mathcal{N}^{\text{Train}}$: Number of train clients; $\mathcal{N}^{\text{Test}}$: Number of test clients; R: Maximum local training rounds; I: Maximum communication rounds; α: Inner learning rate; β: Outer learning rate.

Initialization: θ^{global} (the global model parameters)

PART 1: Local meta-learning in each client

1: for each client $n \in \mathcal{N}^{\text{Train}}$ in parallel do

2: Sample support set $\mathcal{D}_{n,\text{s}}$ and query set $\mathcal{D}_{n,\text{q}}$

3: Initialize the local model θ_n by cloning the global model received from the server

4: for each local training round $\tau = 0,1,\ldots,R-1$ do

5: $\overline{\theta}_{n,\tau+1} \leftarrow \overline{\theta}_{n,\tau} - \alpha \nabla_{\overline{\theta}_{n,\tau}} \mathcal{L}_{\mathcal{D}_{n,\text{s}}} \left(\overline{\theta}_{n,\tau}\right)$, where $\overline{\theta}_{n,0}$ is θ_n

6: τ++, increase the count for the local training round

7: end for

8: $\theta_n \leftarrow \theta_n - \beta \nabla_{\overline{\theta}_{n,R}} \mathcal{L}_{\mathcal{D}_{n,q}} \left(\overline{\theta}_{n,R}\right)$

9: upload θ_n to the server

10: end for

PART 2: Global aggregation in the server

1: for each global communication round $i = 0,\ldots,I-1$ do

2: while all local models are received from the clients do

3: $\theta_{i+1}^{\text{global}} \leftarrow \frac{1}{\mathcal{N}^{\text{Train}}} \sum_{n \in \mathcal{N}^{\text{Train}}} \theta_{i,n}$

4: end while

5: Broadcast the updated global model $\theta_{i+1}^{\text{global}}$ to all participating clients.

6: i++ to increase the count for the global communication round

7: end for

8: Send stop command with the latest global model θ_I^{global} to the clients.

PART 3: Model adaptation in a client

1: Each test client $m \in \mathcal{N}^{\text{Test}}$ downloads the latest global model θ_I^{global} from server

2: Initialize the local model by cloning the global model: $\theta_m \leftarrow \theta_I^{\text{global}}$

3: for each step $h = 0$ to $H-1$ do (only few-shots)

4: Customize the model according to local data: $\theta_{m,h+1} \leftarrow \theta_{m,h} - \alpha \nabla_{\theta_{m,h}} \mathcal{L}_{\mathcal{D}_{\Updownarrow}} \left(\theta_{m,h}\right)$

5: end for

6: Perform personalized analysis based on θ_m

local inner model $\overline{\theta}_{i,n}$ by using local support set $\mathcal{D}_{n,\text{s}}$ as defined in Eq. (15):

$$\overline{\theta}_{i,n,\tau+1} = \overline{\theta}_{i,n,\tau} - \alpha \nabla_{\overline{\theta}_{i,n,\tau}} \mathcal{L}_{\mathcal{D}_{n,\text{s}}} \left(\overline{\theta}_{i,n,\tau}\right), \forall \tau = 0,\ldots,R-1 \tag{15}$$

Note that when τ is 0, $\overline{\theta}_{i,n,0} = \theta_{i,n}$.

After that, the client further updates the local model $\theta_{i,n}$ based on local query set $\mathcal{D}_{n,\text{q}}$ to ensure that the general knowledge is correctly extracted as defined in Eq. (16):

$$\begin{aligned} \theta_{i,n} &= \theta_{i,n} - \beta \nabla_{\theta_{i,n}} \mathcal{L}_{\mathcal{D}_{n,\text{q}}} \left(\overline{\theta}_{i,n,R}\right) \\ &= \theta_{i,n} - \beta \left[\nabla_{\overline{\theta}_{i,n,R}} \mathcal{L}_{\mathcal{D}_{n,\text{q}}} \left(\overline{\theta}_{i,n,R}\right) \frac{\partial \overline{\theta}_{i,n,R}}{\partial \theta_{i,n}}\right] \end{aligned} \tag{16}$$

where $\nabla_{\theta_{i,n}} \mathcal{L}_{\mathcal{D}_{n,\text{q}}} \left(\overline{\theta}_{i,n,R}\right)$ denotes the gradient of the loss function evaluated at the original local model $\theta_{i,n}$, rather than the adapted model $\overline{\theta}_{i,n,R}$ obtained after inner-loop updates. Moreover, it is important to emphasize that the loss function is computed using the query set $\mathcal{D}_{n,\text{q}}$ instead of the support set $\mathcal{D}_{n,\text{s}}$ to guide the outer-loop optimization toward better generalization across tasks.

Then, we further simplify Eq. (16), which involves the second-order derivative, to Eq. (17):

$$\nabla_{\theta_{i,n}} \mathcal{L}_{\mathcal{D}_{n,\text{q}}} \left(\overline{\theta}_{i,n,R}\right) \approx \nabla_{\overline{\theta}_{i,n,R}} \mathcal{L}_{\mathcal{D}_{n,\text{q}}} \left(\overline{\theta}_{i,n,R}\right) \tag{17}$$

Thus, Eq. (16) can be expressed as

$$\theta_{i,n} = \theta_{i,n} - \beta \nabla_{\overline{\theta}_{i,n,R}} \mathcal{L}_{\mathcal{D}_{n,\text{q}}} \left(\overline{\theta}_{i,n,R}\right) \tag{18}$$

After the local model $\theta_{i,n}$ is learned, the client will upload it to the server for global aggregation. Then, the client will wait for further command from the server to either start a new round of local meta-learning or terminate the training to start the model adaptation.

3.4.2 Global aggregation

In this step, the server aggregates local models received from the clients to update the global meta-model θ_i^{global} at communication round i according to Eq. (4). In general, the aggregation weight is set to be equal ($\gamma_n = 1/N$). Alternatively, the weight can be defined as the ratio of the number of samples of client n to the total number of samples.

After the global model aggregation is completed, the server checks the training termination conditions (e.g., maximum communication rounds I). If it is matched, the server will dispatch a stop signal to all the clients and end the learning; otherwise, the server will distribute the updated global meta-model $\theta_{i+1}^{\text{global}}$ to all the clients to start a new global learning round.

3.4.3 Model adaptation

After the global meta-model has been collaboratively trained over the training clients, it is deployed to test clients for personalized adaptation. Let $\mathcal{N}^{\text{Train}}$ and $\mathcal{N}^{\text{Test}}$ denote the sets of clients involved in the training and testing phases, respectively. For each test client $m \in \mathcal{N}^{\text{Test}}$, the goal is to quickly adapt the global meta-model to the local context using a small amount of task-specific data.

The dataset $\mathcal{D}_m$ held by client m may correspond to a newly joined individual in the system or alternatively represent a new behavioral trajectory of an existing user (e.g., change of travel patterns), which is treated as a novel task. In either case, given that the global meta-model θ^{global} contains optimal parameters for a client to adopt, it can be rapidly customized by client m for a localized model θ_m based on its local data $\mathcal{D}_m$ to better support its local context. For each local adaptation step h, the adaptation process is defined as

$$\theta_{m,h+1} = \theta_{m,h} - \alpha \nabla_{\theta_{m,h}} \mathcal{L}_{\mathcal{D}_{\Updownarrow}} \left(\theta_{m,h}\right) \tag{19}$$

In summary, the training process of IPC-FM is described in Algorithm 1. This shows that IPC-FM can not only protect user privacy by processing user sensitive data locally but also support model personalization to alleviate user heterogeneity.

3.5 Model interpretability

To illustrate the model interpretability, we present the mode preference similarity and feature importance displayed by the embedding matrix $\mathbf{W}^{Q}$, as well as the statistical significance calculation procedure of the model parameters.

3.5.1 Mode preference similarity and feature importance

In the proposed T-ANN, we utilize an ANN structure to implement the parameter estimation process of MNL. Additionally, an embedding layer $\mathbf{W}^Q$ is introduced to capture the mapping between personal attributes and available choices. This design enables the relationship between individuals and their corresponding choices to be captured in the embedding space, facilitating the modeling of their similarity. By incorporating the preference coefficients derived from the MNL component, we can further quantify the importance of each personal attribute across different choice alternatives.

Specifically, we first apply L2 normalization to the trained embedding vectors $\mathbf{W}^Q$, which constrains their values within the range $[-1,\ 1]$. This normalization allows the embedding vectors to be treated as unit vectors, enabling cosine similarity to be used to express the degree of association between personal attributes and available choices. Moreover, cosine similarity can also be computed among the embedding vectors of personal attributes to construct an interattribute similarity matrix.

Furthermore, by combining the normalized embeddings (which provide directional information across choice alternatives) with the preference vector $\boldsymbol{\theta}^Q$ (which reflects the strength of preferences), we can obtain a weighted representation that quantifies the importance of each personal attribute for each choice alternative. Notably, during the model training process, we constrain the parameters in $\boldsymbol{\theta}^Q$ to be nonnegative by applying a rectified transformation: $\boldsymbol{\theta}^Q = \{\max\left(0, \theta_1^Q\right), \max\left(0, \theta_2^Q\right), \ldots, \max\left(0, \theta_M^Q\right)\}$. As a result, the signs of the embedding vectors $\mathbf{W}^Q$ will not be influenced by the preference weights, allowing the directionality of the embeddings to independently indicate the nature (positive or negative) of the relationship between individuals and alternative attributes.

3.5.2 Calculation of statistical significance

To verify the effectiveness and interpretability of the model parameters, we conduct statistical significance tests on the coefficient vectors in both the choice property module and the personal attribute module, namely, $\boldsymbol{\theta}^X$ and $\boldsymbol{\theta}^Q$. The evaluation focuses on estimating the standard errors, t-statistics, and p-values for each parameter. Under the assumption that the t-statistics are typically assumed to follow a standard normal distribution. Accordingly, the standard error (SE), t-statistic (t-stat), and p-value for each parameter $\widehat{\theta}_z$ are calculated as

$$\mathrm{SE}\left(\widehat{\theta}_z\right) = \sqrt{(H^{-1})_{zz}} \tag{20}$$

$$t\text{-stat}_z = \frac{\widehat{\theta}_z}{\mathrm{SE}\left(\widehat{\theta}_z\right)} \tag{21}$$

$$p\text{-value}_z = 2\left(1 - \Phi\left(|t\text{-stat}_z|\right)\right) \tag{22}$$

where $\widehat{\theta}$ denotes the trained model parameters, H is the Hessian matrix of the model parameters, and $\Phi(\cdot)$ denotes the CDF of the standard normal distribution.

As is evident from the above, computing statistical significance requires the Hessian matrix of the model parameters. It can be obtained by taking the second-order derivatives of the loss function with respect to the trained model parameters as defined in Eq. (23):

$$H = \frac{\partial^2 \mathcal{L}_{\mathcal{D}}(\theta)}{\partial \theta^2}\Big|_{\theta=\widehat{\theta}} \tag{23}$$

It is worth noting that in this context, only the diagonal elements of the inverse Hessian matrix are needed to estimate the standard errors. Since $\boldsymbol{\theta}^X$ and $\boldsymbol{\theta}^Q$ have relatively limited dimensions, the computational overhead for the Hessian is modest. Hence, it is feasible and practical to employ the Hessian to support significance testing of the model.

4 Experiment

In this section, IPC-FM will be evaluated together with other SOTA methods under the same setting.

4.1 Settings of experiment

This section systematically outlines three critical components of the experiment: (1) data preparation, including data sources, descriptions of variables, and processing methodologies; (2) model training configuration, encompassing both the models included in comparative analysis and their associated hyperparameters; and (3) evaluation metrics utilized for assessing model performance.

4.1.1 Dataset preparation

Two standard datasets on travel mode choice are used, namely, LPMC[1] (Hillel et al., 2018) and Swissmetro (SM)[2] (Bierlaire et al., 2001). Specifically, the LPMC dataset consists of single-day travel diary data obtained by the London Travel Demand Survey from 2012 to 2015. The dataset includes 81,096 samples, each of which corresponds to a trip taken by a person in one of the 17,616 households participating in the survey. The dataset contains four travel modes, namely, walking, cycling, public transportation (pt), and car. To ensure evaluation reliability, households with incomplete information, an insufficient number of choice records, and missing features are excluded. Then, 2975 families were retained. Moreover, the SM dataset comprises passenger survey data gathered in Switzerland. It includes samples from 1291 passengers across nine travel scenarios designed with three kinds of modes, i.e., train, SM, and car. In the travel mode choice scenario, the travel time, departure frequency, and travel costs of each travel mode are defined. Similarly, respondents with incomplete information were excluded. Finally, 974 respondents were retained in the experiment. Note that the key variables of the LPMC and SM datasets used in this study are listed in Tables 1 and 2, respectively. These variables represent standard travel attributes that are widely available in standard surveys and can also be passively collected in many modern mobility service platforms.

After data preprocessing, we treat each family in the LPMC and each individual in the SM as an independent client. Then, we divide these clients into a training set (i.e., $\mathcal{N}^{\mathrm{Train}}$) and a testing set (i.e., $\mathcal{N}^{\mathrm{Test}}$) according to the proportions in Table 3. Furthermore, each client partitions its own data into support and query sets with a ratio of 7 to 3.

4.1.2 Model training configuration

The mode of model training can be divided into centralized and decentralized. For the centralized mode, we consider the following four models:

- **MNL**: An ANN-based MNL implementing the estimation process of MNL through ANN;
- **ASU-DNN** (Wang et al., 2020a): A DNN model with alternative-specific utility functions;

[1] https://www.emerald.com/jsmic/article-supplement/408759/csv/dataset/

[2] https://biogeme.epfl.ch/#data

Table 1 Variable description of the LPMC dataset

Variable	Description
HOUSEHOLD ID	Identifier of the household
PURPOSE	Travel purpose (i.e., B: business, HBE: home-based education, HBO: home-based other, HBW: home-based work, NHBO: non-home-based other)
FUELTYPE	Fuel type of the vehicle (diesel/petrol/hybrid car or diesel/petrol *LGV)
FARETYPE	Public transport fare type
BUS SCALE	Percentage of the bus fare paid by the traveler
DAY OF WEEK	The day of week for a trip
AGE	Age group (i.e., 20–39, 40–59, 60–79)
FEMALE	Gender of the traveler
DRIVING LICENSE	Whether the traveler holds a driving license
CAR OWNERSHIP	Car ownership of the household (i.e., no cars, <1 car per adult, ≥ 1 car per adult)
DISTANCE	Straight line travel distance (km)
INTERCHANGES NUM	Total number of public transport interchanges (e.g., rail-rail, bus-bus, bus-rail, rail-bus)
START TIME	The trip starting period (i.e., 0–5, 6–11, 12–17,18–23)
MONTH	Month of the year for a travel
YEAR	The year for a travel
DURATION	Travel duration (h)
COST	Estimated travel cost (GBP)
TRAVEL MODE	The travel mode (i.e., walking, cycling, pt, car)

Note: LGV-light goods vehicle.

Table 2 Variable description of the SM dataset

Variable	Description
ID	Traveler identifier
PURPOSE	Trip purpose (i.e., commuter, shopping)
FIRST	First class traveler
TICKET	Ticket type (i.e., Annual_junior_senior, Free_after_7pm, and Other_tic.)
WHO	Who pays the ticket (i.e., self, employer, unknown)
MALE	Traveler gender
LUGGAGE	Luggage number (i.e., none, one piece, several pieces)
AGE	Age group (i.e., 15–24, 25–39, 40–54)
INCOME	Income class per year [thousand CHF] (i.e., less than 50, 50–100, over 100)
GA	Traveler with annual season tickets
SEATS	Seats configuration in the SM (dummy)
ORIGIN	Travel origin
DEST	Travel destination
TT	Door-to-door travel time of different travel modes (min), scaled by 1/100
TC	Travel costs (CHF), scaled by 1/100
HEADWAY	Headway (min), scaled by 1/100
CHOICE	Travel mode (i.e., Train, SM, Car)

- **E-MNL** (Arkoudi et al., 2023): An ANN-based MNL with discrete individual features mapped through the embedding layer;
- **L-MNL** (Sifringer et al., 2020): A model combining the ANN-based MNL with a dense neural network

For the decentralized mode, based on the proposed T-ANN model, two approaches are used, namely:

- **IPC-FL**: The proposed method is only configured to run FL, in which clients will train a general model (instead of a meta-model) based on its local data;
- **IPC-FM**: The proposed method runs FL and meta-learning, in which clients train the meta-model.

During model training, the two aforementioned datasets are employed to simulate continuous data collection processes, mirroring realistic application scenarios in which user data are persistently gathered and transmitted. In the centralized training mode, training clients periodically upload raw data to a central server for unified processing. In contrast, the decentralized mode adopts a federated learning framework, where clients retain raw data locally and only transmit model parameters to the server, thereby enhancing user privacy. For both training paradigms, the support and query sets of each raining client are used jointly to train the models. During performance evaluation, the trained models are first locally adapted using the support set of the test clients and then evaluated on their corresponding query sets. To make a fair comparison and ensure the reproducibility of the model training, we use hyperparameters with optimal performance, as listed in Table 3. This training process does not employ the early stopping mechanism. Each model is trained for a fixed number of epochs. Since each client processes only a small number of records and the server performs simple parameter aggregation, the computational cost of both local updates and

Table 3 Hyperparameters used by the compared models

Hyperparameter	Value
Invariant hyperparameter	
Activation function	ReLU and Softmax
Initialization	Glorot
Optimizer	Stochastic gradient descent (SGD)
Loss function	Cross-entropy
Batch size (centralized)	200
Batch size (FL and FM)	Individual data
Number of training rounds R (FL and FM)	1
Varying hyperparameter	
L1 regularization	[10^{-7}, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}]
L2 regularization	[10^{-7}, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}]
Learning rate	[0.001, 0.002, 0.005, 0.01, 0.02]
Epochs	[100, 200, 400, 800, 1600]
Dropout rate	[0.001, 0.01, 0.1, 0.3, 0.5]
Noise level	[0.01, 0.05, 0.1, 0.15, 0.2]
Miss rate	[0.01, 0.05, 0.1, 0.15, 0.2]
Proportion of clients for testing	[0.30, 0.35, 0.40, 0.45, 0.50]

global coordination remains modest. All experiments are conducted on a workstation equipped with 1 NVIDIA GeForce RTX 3060 GPU and 64 GB RAM. This setup is used to simulate the interaction between clients and the server. Note that the source code can be downloaded from link[3].

4.1.3 Evaluation metrics

The following metrics are used in the experiment:

- Accuracy: The average accuracy;
- Test loss: It is measured by cross-entropy;
- F1 score: This score combines precision and recall to evaluate the overall performance;
- Kappa score: The kappa score is used to evaluate the interrater reliability of the model.

Specifically, the accuracy, test loss, F1 score, and Kappa score are calculated by

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{24}$$

$$\text{Loss} = -\sum_{n=1}^{N}\sum_{j=1}^{J} y_{n,j}\log\left(P_{n,j}\right) \tag{25}$$

$$\text{F1} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{26}$$

$$\text{Kappa} = \frac{p_o - p_e}{1 - p_e} \quad \text{s.t.}$$

$$p_e = \frac{a_1 \cdot b_1 + a_2 \cdot b_2 + \cdots + a_N \cdot b_N}{N^2} \tag{27}$$

where TP, TN, FP, and FN represent true positive, true negative, false positive, and false negative, respectively; $y_{n,j}$ is a binary variable, when the individual n selects the alternative j, its value is 1 and otherwise 0; Precision and recall are, respectively defined as $TP/(TP+FP)$ and $TP/(TP+FN)$; p_o represents the average accuracy of all classes, and a_j and b_j are the number of real and predicted samples for each category, respectively.

4.2 Evaluation results

The evaluation consists of three parts: (1) the performance of centralized and decentralized methods; (2) the robustness of the method; and finally, (3) the interpretability of the method.

4.2.1 Performance of the method

As listed in Table 4, IPC-FM outperforms the other methods. Specifically, compared with the basic MNL model, in the LPMC dataset, the accuracy, F1 score, Kappa score, and test loss of IPC-FM increase by 16%, 15%, 34%, and 41%, respectively, and in the SM dataset, related metrics also increase by 8%, 20%, 18%, and 25%, respectively. This indicates that without collecting and processing sensitive user data at the server, the proposed privacy-preserving method IPC-FM can train a better performing model than centralized methods after 1 step adaptation. During the model adaptation phase, as shown in Table 5, IPC-FM can achieve better performance after a few steps of local adaptation. Specifically, in the LPMC dataset, the global model of IPC-FM (i.e., no adaptation) achieved an initial accuracy of 66.12%. After just 4 steps of local fine-tuning, the accuracy improves significantly to 81.68%, representing a 15.56% enhancement. Notably, IPC-FM consistently outperforms all compared models at each adaptation step. Similar improvements are observed on the SM dataset, where the global model initially showed 61.04% accuracy. After local adaptation, the accuracy exceeded 73%, marking an 11.96% improvement and maintaining superior performance across all comparative methods. These results demonstrate that IPC-FM can rapidly adapt to different user contexts while sustaining performance advantages over competing methods.

4.2.2 Robustness of the method

As illustrated in Fig. 5, the impact of regularization (L1 and L2) on the accuracy is analyzed. Regardless of different regularization

Table 4 Performance assessment by using the test set of the LPMC and SM datasets

Model	LPMC				SM			
	Accuracy	F1 score	Kappa	Loss	Accuracy	F1 score	Kappa	Loss
MNL	63.36%	41.78%	33.33%	2965.26	67.58%	46.91%	39.03%	727.43
ASU-DNN	73.31%	53.58%	55.74%	2183.15	73.15%	54.90%	49.99%	627.28
E-MNL	71.62%	51.52%	51.79%	2281.22	72.01%	56.57%	48.41%	629.93
L-MNL	72.36%	52.99%	53.24%	2285.95	72.13%	53.81%	48.13%	621.27
IPC-FL	76.12%	55.68%	59.92%	2053.40	73.27%	54.18%	50.17%	606.40
IPC-FM	**79.78%**	**66.66%**	**66.91%**	**1755.40**	**75.77%**	**69.62%**	**57.05%**	**545.48**

Note: **Bold** numbers indicate the best performance. Numbers underlined are the second-best performance.

[3] https://github.com/IntelligentSystemsLab/IPC-FM

configurations, IPC-FM can retain the highest accuracy for both the LPMC and SM datasets. Specifically, in LPMC, the accuracy of IPC-FM can fluctuate by less than 2%, even when other methods experience a significant performance drop after the L1/L2 penalty becomes larger than 10^{-3}. Similarly, in SM, compared to other methods, IPC-FM can also retain the best performance with a more stable accuracy curve.

Moreover, Figs. 6a–6c and 7a–7c demonstrate the effect of the learning rate, epochs, and dropout on the model performance. It can be seen that in general, the increase in the learning rate and epochs has less impact on the model performance compared to the increase in the dropout rate. Despite the changes in these hyperparameters, ICP-FM can remain superior among all compared methods with less vibration in accuracy. Such a result reveals the efficiency and robustness of the proposed model T-ANN designed and implemented in IPC-FM.

Furthermore, we assess the performance under varying levels of noise and incomplete data to simulate real-world data imperfections. As shown in Figs. 6d and 6e and Figs. 7d and 7e, IPC-FM consistently maintains the highest accuracy and the most stable performance curves under both Gaussian noise with levels defined by the standard deviation for continuous variables and

Table 5 Accuracy after local adaptation for the LPMC and SM datasets

Dataset	Model	No adaptation	Adaptation step 1	Adaptation step 2	Adaptation step 3	Adaptation step 4
LPMC	MNL	63.24±0.00%	63.36±0.00%	63.36±0.00%	63.33±0.00%	63.51±0.00%
	ASU-DNN	**71.27±0.31%**	72.81±0.30%	74.18±0.32%	75.23±0.30%	76.11±0.34%
	E-MNL	70.72±0.57%	71.07±0.56%	71.38±0.55%	71.72±0.54%	72.03±0.55%
	L-MNL	70.48±0.27%	71.63±0.28%	72.58±0.24%	73.42±0.25%	74.19±0.22%
	IPC-FL	69.29±0.73%	73.37±1.92%	75.10±2.02%	76.38±2.03%	77.13±1.88%
	IPC-FM	66.12±1.63%	**78.75±0.53%**	**80.66±0.70%**	**81.38±0.81%**	**81.68±0.86%**
SM	MNL	67.15±0.11%	69.26±1.39%	69.68±1.53%	70.13±1.69%	70.51±1.87%
	ASU-DNN	**69.68±0.70%**	70.22±0.71%	71.24±0.59%	72.05±0.53%	72.80±0.61%
	E-MNL	68.98±1.38%	69.34±1.55%	69.61±1.59%	69.88±1.63%	70.11±1.61%
	L-MNL	68.82±0.48%	69.59±0.46%	70.32±0.62%	71.14±0.63%	71.84±0.59%
	IPC-FL	68.12±1.33%	70.24±1.56%	71.17±1.80%	71.70±1.86%	72.39±2.03%
	IPC-FM	61.04±2.30%	**73.02±1.12%**	**74.21±1.27%**	**74.20±1.24%**	**73.65±1.34%**

Note: **Bold** numbers indicate the best performance. Numbers underlined are the second-best performance.

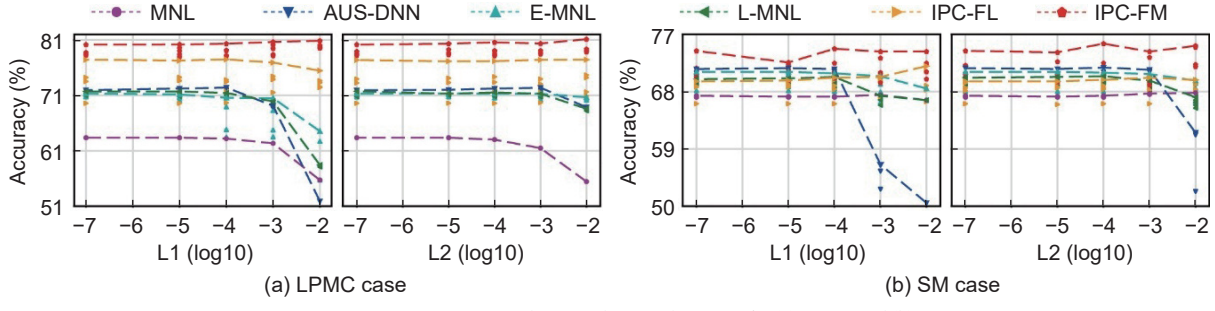


Fig. 5 Accuracy comparison under L1 and L2 regularization for (a) LPMC and (b) SM.

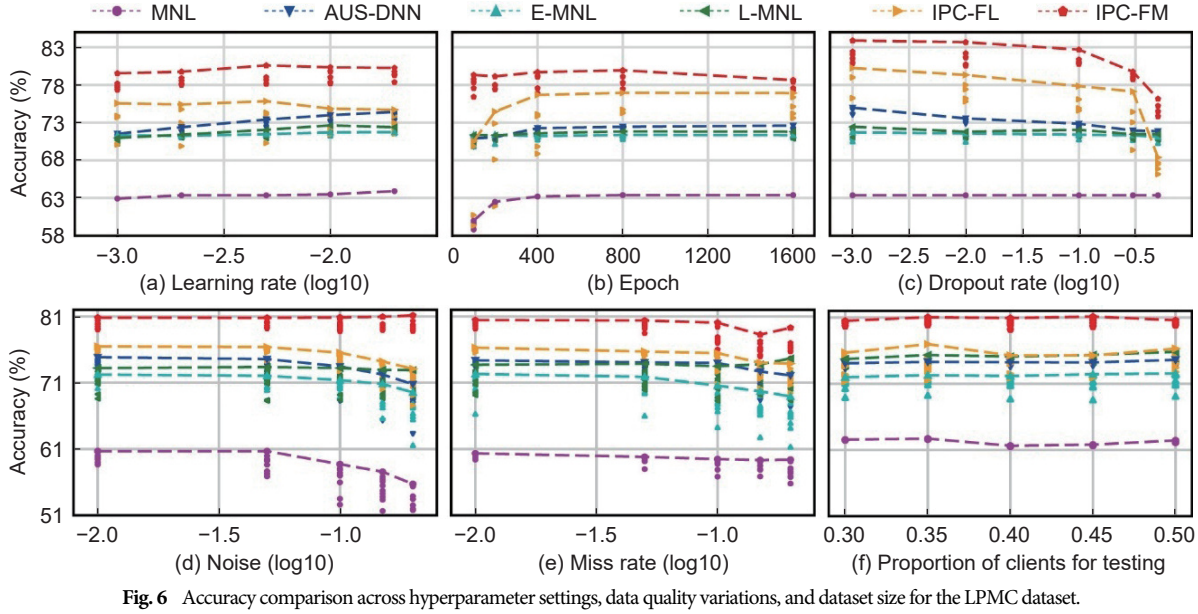


Fig. 6 Accuracy comparison across hyperparameter settings, data quality variations, and dataset size for the LPMC dataset.

discrete label corruption with levels specifying the proportion of flipped labels for attributes such as gender across both the LPMC and SM datasets. This result strongly demonstrates that the IPC-FM framework exhibits superior resilience to data quality issues, further reinforcing its robustness against external disturbances.

Finally, to assess the model performance under different dataset sizes, we examine the impact of varying the proportion of clients assigned to the testing set. This setting reduces the number of clients available for training and allows us to investigate how the performance changes when the training data become more limited. As shown in Figs. 6f and 7f, IPC-FM consistently achieves the highest accuracy across a wide range of testing proportions for both datasets. Its accuracy remains relatively stable even when the testing proportion increases from 0.3 to 0.5, which leaves fewer clients for model training. In contrast, the other methods display

more noticeable accuracy drops as the training data decrease. These results demonstrate that IPC-FM is robust to reductions in training data and can maintain reliable performance in data-limited scenarios.

4.2.3 Interpretability of the method

Given the criticism of "black-box" DNNs, post hoc explanations are required to analyze the choice probabilities (Wang et al., 2020a; Martín-Baos et al., 2023). First, Fig. 8 presents the probabilities of the compared models in analyzing the choice behavior of the four travel modes defined in the LPMC when driving costs escalate. It is worth noting that the dark line in Fig. 8 represents the overall choice probability curve across clients. By holding other factors constant, the curve can provide insights about the response of different models to changes in driving costs. Accordingly, it is observable that ASU-DNN, IPC-FL and IPC-FM have similar results to MNL in the overall choice probability. However, in the overall probability curve of the ASU-DNN, there exist regions in which a higher travel cost is associated with a higher probability of choosing a private car, which makes the result less reasonable. In addition, for the choice probability curve of a client (the bright line), compared with MNL and IPC-FL, IPC-FM is more diversified, showing its ability to support more personalized analysis. Moreover, these curves can be directly interpreted at the level of individual records by analyzing how changes in the attributes of a specific user influence their predicted choice probabilities, thus providing individual-level interpretability. Such a result illustrates that IPC-FM is capable of capturing more intuitive, reasonable and user-oriented choice probabilities. Although some individuals show nearly fixed choice probabilities (e.g., consistently high walking probability), this does not indicate model overfitting. The adaptation in IPC-FM is based on a few steps from a meta-learned initialization, making the model robust to overfitting even when local data are limited or imbalanced. These patterns are likely due to the underlying distribution of user behavior in the support set.

Second, as listed in Table 6, the linear parameters, such as choice properties and personal attributes, are analyzed to illustrate the significance of different features. We can see that all variables are statistically significant (p-value < 0.01), and both "DURATION" and "COST" produce negative signals, which are

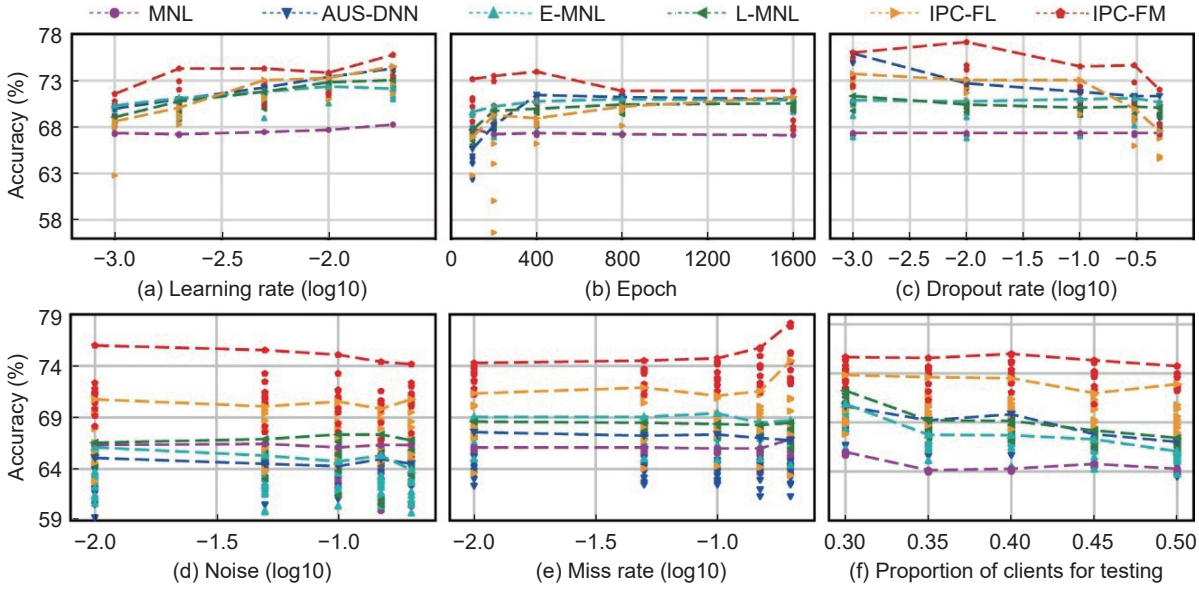


Fig. 7 Accuracy comparison across hyperparameter settings, data quality variations, and dataset size for the SM dataset.

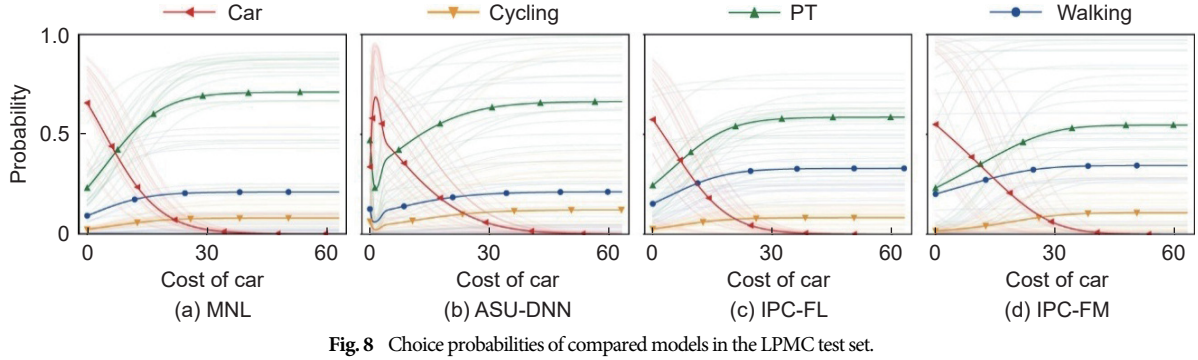


Fig. 8 Choice probabilities of compared models in the LPMC test set.

Table 6 Parameter estimation of IPC-FM on the LPMC dataset

Variable	θ^X & θ^Q	SE	*t*-stat	*p*-value
ASC_CYCLE	−1.471	0.053	−27.525	0.000
ASC_PT	0.322	0.028	11.307	0.000
ASC_CAR	0.528	0.037	14.087	0.000
DURATION	−5.627	0.106	−53.129	0.000
COST	−0.127	0.008	−16.539	0.000
DAY OF WEEK	0.589	0.072	8.173	0.000
AGE	0.667	0.127	5.238	0.000
FEMALE	0.701	0.063	11.165	0.000
DRIVING LICENSE	0.526	0.043	12.222	0.000
BUS SCALE	0.439	0.106	4.152	0.000
CAR OWNERSHIP	0.772	0.048	16.080	0.000
FARETYPE	0.814	0.073	11.215	0.000
PURPOSE	0.834	0.069	12.139	0.000
FUELTYPE	0.665	0.084	7.919	0.000
START TIME	0.408	0.083	4.906	0.000
TRAVEL MONTH	0.707	0.086	8.203	0.000
TRAVEL YEAR	0.421	0.134	3.136	0.002
PT INTERCHANGES	0.387	0.099	3.895	0.000
DISTANCE	1.771	0.055	32.055	0.000

consistent with the theoretical expectations. This shows that IPC-FM can retain the interpretability of conventional models, such as DCMs. It is worth noting that the parameters listed in Table 6 represent the linear components of the utility function, specifically derived from the choice property and personal attribute modules. Although personal attributes may influence utility through both linear and nonlinear terms, the interpretation of linear parameters is still meaningful. This design enables the model to maintain interpretability similar to DCMs while benefiting from the data mining capabilities of deep learning.

Third, Fig. 9 shows the association direction between embedding vectors of personal attributes and available travel modes. The value in each embedding dimension indicates the dependency of choice related to a given feature, e.g., it can be observed that features positively correlated with Car (with value close to 1 in the direction of the Car axis) are travelers older than 81 and those with car type diesel_LGV; in contrast, features negatively correlated with Car include travelers who do not own a car or have cars with fuel type Average_car.

These results further reveal the interpretability of IPC-FM in analyzing user choice behaviors.

Finally, the embedding vectors are normalized in a fixed range $[-1,\ 1]$, $f^Q(Q_n)$, which shows the utility of personal attributes. The global model can be used to analyze the overall impact of a feature on a choice, i.e., how and to what extent a given feature affects a given choice. The individual model can also be used to analyze the impact of this feature on individual travel, explaining each data record. Specifically, as shown in Fig. 10, we can visually analyze the impact of each feature on the choice of travel modes (i.e., walking, cycling, pt, and car) and determine the profile of travelers, who show high and low preferences for each choice, e.g., people who travel for B (business), are strongly averse to Walking, and travel for HBE (home-based education) and NHBO (non home-based other) are reluctant to bikes. In addition, it is intuitive to identify preferable groups in each travel mode, i.e., (1) those aged approximately 21–40 tend to walk, (2) those who travel for HBW (home-based work) are inclined to cycle, (3) those aged 61–80 prefer to use PT, and (4) those older than 80 prefer to travel by Car. Unsurprisingly, the result also accurately reveals that elderly people in the 61–80 age group, who, in general, retain better mobility than the traveler group older than 80, favor PT more, as local residents older than 60 can travel freely on buses, tubes and other public transport in London.

In summary, the above analysis highlights the efficiency and effectiveness of IPC-FM, which can not only address heterogeneous user data in a privacy-preserving manner for an outperforming and robust model but also show strong model interpretability to produce reasonable and intuitive results and insights for user behavior analysis.

5 Conclusions and discussion

This study addresses the challenges faced by applying DNNs in supporting travel behavior analysis in a privacy-preserving and user-oriented context. In contrast to centralized methods, the proposed IPC-FM method first designs the T-ANN as the interpretable model to extract both linear and nonlinear relationships encoded in user data and then addresses user heterogeneity by T-MUP to

collaboratively train an adaptive model that can be rapidly localized to support personalized analysis.

The experimental results show that IPC-FM is able to train a global meta-model that can be rapidly adopted by users to support analysis under heterogeneous contexts, and compared to the second-best method, IPC-FM can significantly improve accuracy by 3%, test loss by 10%, F1 score by 11%, and kappa score by 7%. Meanwhile, IPC-FM demonstrates rapid adaptation capability, which cannot improve accuracy by approximately 12% within 4 adaptation steps but also consistently outperforms all competing

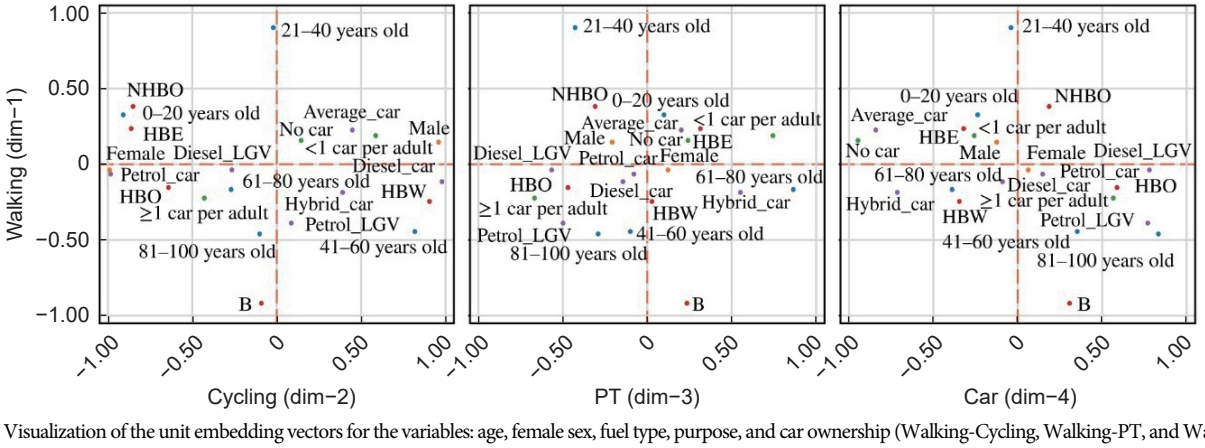


Fig. 9 Visualization of the unit embedding vectors for the variables: age, female sex, fuel type, purpose, and car ownership (Walking-Cycling, Walking-PT, and Walking-CAR axes pairs).

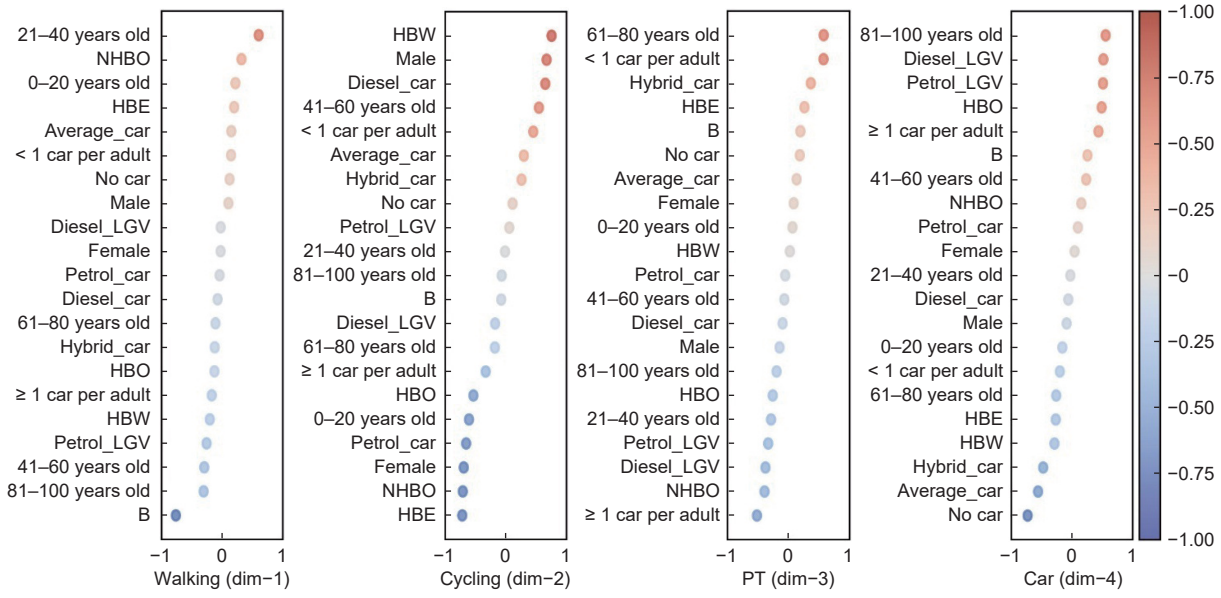


Fig. 10 Visualization of the utility of the personal attributes for each choice alternative.

methods at each step. It is important to emphasize that IPC-FM is designed with the expectation that local adaptation will be performed before deployment. Although the global meta-model can be used without adaptation, this is suboptimal and may limit its ability to capture individual preferences. In addition, under different model configurations in terms of regularization methods, learning rate, epoch, and dropout rate, as well as noise and missing values, IPC-FM can consistently maintain the highest performance in all cases, showing its robustness in training a stable and outperforming model. Furthermore, the T-ANN designed and trained within IPC-FM can provide intuitive and interpretable results, based on which, the reasons behind user choices can be better analyzed.

From the above experimental results, IPC-FM demonstrates excellent performance in scenarios where limited user data can be collected locally. In addition, IPC-FM also demonstrates promising potential for practical deployment. First, data collection reduces the need for fully traditional surveys or app-based collection. Instead, user data can be passively and continuously collected through service-oriented applications and then processed at the local device. The design of IPC-FM fully accommodates the future incorporation of passively collected data due to its distributed, privacy-preserving learning process and its novel utility function structure for capturing nonlinear feature impacts. Next, IPC-FM can enable cross-agency, cross-regional, and cross-institutional knowledge sharing. Without the need to exchange raw data, stakeholders can collaboratively train a shared meta-model via FM, which can then be personalized locally to meet the specific demands of cities, institutions, or regions. Finally, beyond the fundamental privacy protection offered by keeping data on local devices, IPC-FM introduces an additional model-level safeguard through its dual-level meta-training scheme. By separating the support set (for adaptation) and the query set (for meta-update), the framework reduces the granularity of information contained in the shared gradient updates. This design inherently complicates potential gradient-based privacy attacks, thereby offering a practical layer of security enhancement over standard federated learning. Therefore, IPC-FM is particularly suitable for deployment in intelligent transportation systems, mobility services, and recommendation platforms that require personalization, collaboration, and data protection. In such scenarios, a small amount of recent user interaction data is available through passive collection, and this information can serve as a support set to facilitate local adaptation of the IPC-FM model. When combined with simulation platforms such as SimMobility (Zhu et al., 2018), the behavioral analysis results generated by IPC-FM can support planners in evaluating the accessibility of different areas and improving short-term travel demand forecasting and path optimization. These short-term outcomes can further inform the assessment of middle-term accessibility patterns, which in turn influence long-term household and firm location choices. This multilevel connection enables urban planners to understand the spatial implications of mobility policies and supports integrated land use and transportation planning (Zhao et al., 2025). In addition, interpretable utility components can support urban planners in analyzing the influence of travel time, cost, sociodemographic factors, and latent behavior patterns. These insights can assist in policy evaluation, demand analysis, and scenario assessment in practical tasks.

Despite these advances, as an initial attempt to support user behavior analysis in decentralized and data-preserving scenarios, this study still faces challenges in optimizing frequent client-server interactions during the model training process to impel the transfer of key knowledge extracted from local data. In the future, it is worthwhile to investigate a dedicated training mechanism that can achieve a balance between learning cost and model performance. Moreover, in real-world scenarios, the availability of personal devices may change over time and places, and asynchronous collaboration among users can be explored to address issues caused by stragglers. It is also important to acknowledge that the current approach does not incorporate formal privacy guarantees such as DP. Future work could explore integrating noise-based mechanisms or secure aggregation protocols to quantify and enhance the level of privacy protection in federated meta-learning. Last, the potential of leveraging large models, such as large language models, to support user behavior analysis can be studied to provide a common foundation for personalized services in intelligent transportation systems.

Author contributions

Linlin You: Funding acquisition, Project administration, Supervision, Visualization, Writing – Original Draft. **Kunxu Chen**: Conceptualization, Data Curation, Formal Analysis, Methodology, Validation, Visualization, Writing – Original Draft,

Writing – Review & Editing. **Baichuan Mo**: Formal analysis, Validation, Writing – Original Draft. **Jiemin Xie**: Investigation, Supervision, Project administration. **Juanjuan Zhao**: Validation, Project administration, Funding acquisition, Writing – review & editing. **Jinhua Zhao**: Validation, Supervision, Resources.

Replication and data sharing

The source codes and replication package are available on ETS data at https://doi.org/10.26599/ETSD.2026.9190002.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (No. 62576366), the National Key R&D Program of China (No. 2023YFB4301900), the Guangdong Basic and Applied Basic Research Foundation (No. 2023A1515012895), and the Department of Science and Technology of Guangdong Province (No. 2021QN02S161).

Declaration of competing interest

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

References

Acar, D. A. E., Zhao, Y., Navarro, R. M., Mattina, M., Whatmough, P. N., Saligrama, V., 2021. Federated Learning Based on Dynamic Regularization. https://doi.org/10.48550/arXiv.2111.04263.

Arkoudi, I., Krueger, R., Azevedo, C. L., Pereira, F. C., 2023. Combining discrete choice models and neural networks through embeddings: Formulation, interpretability and performance. *Transp Res Part B Methodol*, **175**, 102783.

Bierlaire, M., Axhausen, K., Abay, G., 2001. The acceptance of modal innovation: The case of Swissmetro. In: the 1st Swiss Transport Research Conference, 1–15.

Chauhan, R.S., Sutradhar, U., Rozhkov, A., Derrible, S., 2023. Causation Versus Prediction: Comparing Causal Discovery and Inference with Artificial Neural Networks in Travel Mode Choice Modeling. https://doi.org/10.48550/arXiv.2307.15262

Cunha, J. F. E., Galvão, T., 2014. State of the art and future perspectives for smart support services for public transport. In: Service Orientation in Holonic and Multi-Agent Manufacturing and Robotics, 225–234.

Elharoun, M., El-Badawy, S. M., Shahdah, U. E., 2023. Artificial intelligence techniques for predicting individuals' mode choice behavior in mansoura city, Egypt. *Transp Res Board*, **2677**, 605–623.

Fallah, A., Mokhtari, A., Ozdaglar, A., 2020. Personalized federated learning with theoretical guarantees: a model-agnostic meta-learning approach. In: Advances in Neural Information Processing Systems. Curran Associates, 3557–3568.

Finn, C., Abbeel, P., Levine, S., 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In: International Conference on Machine Learning, 1126–1135.

Gao, L., Fu, H., Li, L., Chen, Y., Xu, M., Xu, C. Z., 2022. FedDC: Federated learning with non-IID data via local drift decoupling and correction. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 10102–10111.

Hillel, T., Elshafie, M. Z. E. B., Jin, Y., 2018. Recreating passenger mode choice-sets for transport simulation: A case study of London, UK. *Proc Inst Civ Eng Smart Infrastruct Constr*, **171**, 29–42.

Hospedales, T., Antoniou, A., Micaelli, P., Storkey, A., 2022. Meta-learning in neural networks: A survey. *IEEE Trans Pattern Anal Mach Intell*, **44**, 5149–5169.

Jian, W., He, J., Chen, K., Xie, J., Zhao, J., You, L., 2025. Personalized travel recommendations based on asynchronous and privacy-preserving mixed logit model. *IEEE Trans Intell Transp Syst*, **26**, 2227–2238.

Kamal, K., Farooq, B., 2024. A Deep Causal Inference Model for Fully Interpretable Travel Behavior Analysis. https://doi.org/10.48550/arXiv.2405.01708

Li, W., Feng, W., Yuan, H. Z., 2020. Multimode traffic travel behavior characteristics analysis and congestion governance research. *J Adv Transp*, **2020**, 6678158.

Li, X., Huang, K., Yang, W., Wang, S., Zhang, Z., 2019. On the Convergence of FedAvg on Non-IID Data. https://doi.org/10.48550/arXiv.1907.02189

Li, X., Li, Y., Wang, J., Chen, C., Yang, L., Zheng, Z., 2022. Decentralized federated meta-learning framework for few-shot multitask learning. *Int J Intelligent Sys*, **37**, 8490–8522.

Lin, G., Qian, S., Khattak, Z. H., 2025. FedAV: Federated learning for cyberattack vulnerability and resilience of cooperative driving automation. *Commun Transp Res*, **5**, 100175.

Liu, S., You, L., Zhu, R., Liu, B., Liu, R., Yu, H., et al., 2024. AFM3D: An asynchronous federated meta-learning framework for driver distraction detection. *IEEE Trans Intell Transp Syst*, **25**, 9659–9674.

Ma, D., Song, X., Li, P., 2021. Daily traffic flow forecasting through a contextual convolutional recurrent neural network modeling inter- and intra-day traffic patterns. *IEEE Trans Intell Transp Syst*, **22**, 2627–2636.

Martín-Baos, J. Á., López-Gómez, J. A., Rodriguez-Benitez, L., Hillel, T., García-Ródenas, R., 2023. A prediction and behavioural analysis of machine learning methods for modelling travel mode choice. *Transp Res Part C Emerg Technol*, **156**, 104318.

McFadden, D., 1974. The measurement of urban travel demand. *J Public Econ*, **3**, 303–328.

McMahan, B., Moore, E., Ramage, D., Hampson, S., y Arcas, B.A., 2017. Communication-efficient learning of deep networks from decentralized data. In: Artificial Intelligence and Statistics, 1273–1282.

Moreau, H., Vassilev, A., Chen, L., 2022. The devil is in the details: An efficient convolutional neural network for transport mode detection. *IEEE Trans Intell Transp Syst*, **23**, 12202–12212.

Park, K., Reisinger, Y., 2010. Differences in the perceived influence of natural disasters and travel risk on international travel. *Tour Geogr*, **12**, 1–24.

Qu, H., Liu, S., Li, J., Zhou, Y., Liu, R., 2022. Adaptation and learning to learn (ALL): An integrated approach for small-sample parking occupancy prediction. *Mathematics*, **10**, 2039.

Salih, A. M., Galazzo, I. B., Gkontra, P., Rauseo, E., Lee, A. M., Lekadir, K., et al., 2024. A review of evaluation approaches for explainable AI with applications in cardiology. *Artif Intell Rev*, **57**, 240.

Sethu, S. G., 2020. Legal protection for data security: A comparative analysis of the laws and regulations of European union, US, India and UAE. In: 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT), 1–5.

Sifringer, B., Lurkin, V., Alahi, A., 2020. Enhancing discrete choice models with representation learning. *Transp Res Part B Methodol*, **140**, 236–261.

Sundararajan, M., Najmi, A., 2020. The many Shapley values for model explanation. In: International Conference on Machine Learning, 9269–9278.

Train, K., 2009. *Discrete Choice Methods with Simulation*. New York: Cambridge University Press.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., et al., 2017. Attention is all you need. *Advances in Neural Information Processing Systems*, **30**, 5998–6008.

Wang, S., Mo, B., Zhao, J., 2020a. Deep neural networks for choice analysis: Architecture design with alternative-specific utility functions. *Transp Res Part C Emerg Technol*, **112**, 234–251.

Wang, S., Mo, B., Zheng, Y., Hess, S., Zhao, J., 2024. Comparing hundreds of machine learning and discrete choice models for travel demand modeling: An empirical benchmark. *Transp Res Part B Methodol*, **190**, 103061.

Wang, S., Wang, Q., Bailey, N., Zhao, J., 2021. Deep neural networks for choice analysis: A statistical learning theory perspective. *Transp Res Part B Methodol*, **148**, 60–81.

Wang, S., Wang, Q., Zhao, J., 2020b. Deep neural networks for choice analysis: Extracting complete economic information for interpretation. *Transp Res Part C Emerg Technol*, **118**, 102701.

Wong, M., Farooq, B., 2021. ResLogit: A residual neural network logit model for data-driven choice modelling. *Transp Res Part C Emerg Technol*, **126**, 103050.

Yang, Q., Liu, Y., Chen, T., Tong, Y., 2019. Federated machine learning: Concept and applications. *ACM Trans Intell Syst Technol*, **10**, 1–19.

You, L., Guo, Z., Yuen, C., Chen, C. Y., Zhang, Y., Poor, H. V., 2025. A framework reforming personalized Internet of Things by federated meta-learning. *Nat Commun*, **16**, 3739.

You, L., Hao, M., Sun, J., Wang, Y., Rong, C., Yuen, C., et al., 2024. Toward a personalized autonomous transportation system: Vision, challenges, and solutions. *Innovation*, **5**, 100704.

You, L., Liu, S., Chang, Y., Yuen, C., 2022. A triple-step asynchronous federated learning mechanism for client activation, interaction optimization, and aggregation enhancement. *IEEE Internet Things J*, **9**, 24199–24211.

You, L., Zhao, F., Cheah, L., Jeong, K., Zegras, C., Ben-Akiva, M., 2018. Future mobility sensing: An intelligent mobility data collection and visualization platform. In: 2018 21st International Conference on Intelligent Transportation Systems (ITSC), 2653–2658.

Zhao, J., Mo, B., Caros, N. S., Zhao, J., 2025. Housing exchange framework to reduce carbon emissions from commuting. *Nat Sustain*, **8**, 1259–1269.

Zhao, X., Yan, X., Yu, A., Van Hentenryck, P., 2020. Prediction and behavioral analysis of travel mode choice: A comparison of machine learning and logit models. *Travel Behav Soc*, **20**, 22–35.

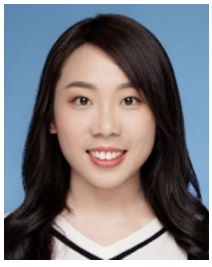
Zhu, Y., Diao, M., Ferreira, J., Zegras, C., 2018. An integrated microsimulation approach to land-use and mobility modeling. *J Transp Land Use*, **11**, 633–659.

Linlin You is an Associate Professor at the School of Intelligent Systems Engineering, Sun Yat-sen University, and a Research Affiliate at the Intelligent Transportation System Lab, Massachusetts Institute of Technology. He was a Senior Postdoc at the Singapore-MIT Alliance for Research and Technology and a Research Fellow at the Architecture and Sustainable Design Pillar of Singapore University of Technology and Design. He received the Ph.D. degree in computer science from the University of Pavia in 2015. He has published more than 100 journal and conference papers in the research fields of smart cities, multisource data fusion, machine learning, and federated learning. He is an Associate Editor of *SN Computer Science*, Editorial Board Member of *Scientific Reports*, and Youth Editor of *The Innovation*.

Kunxu Chen is currently pursuing the M.S. degree in the School of Intelligent Systems Engineering, Sun Yat-sen University. He received the B.S. degree in traffic engineering from Sun Yat-sen University in 2024. His research interests include machine learning, federated learning, analysis of travel behaviors, and their applications in intelligent transportation systems.

Baichuan Mo received the B.S. degree in the civil engineering from Tsinghua University, the M.S. degree in transportation and computer science, and the Ph.D. degree in transportation from Massachusetts Institute of Technology (MIT) in 2022. He is currently a Research Scientist with Lyft. His research interests include data-driven transportation modeling, demand modeling, optimization, and applied machine learning.

Jiemin Xie is currently an Assistant Professor with the School of Intelligent Systems Engineering, Sun Yat-sen University. She received the Ph.D. degree in transportation engineering from the University of Hong Kong, Hong Kong, China in 2020. She published 26 journal and conference papers in the research fields of smart transportation, passenger behavior modeling, and transit planning.



Juanjuan Zhao is currently an Assistant Professor at the College of Resources Environment and Tourism, Capital Normal University. She was a Post-Doctoral Researcher at the Singapore-MIT Alliance for Research and Technology. She received the Ph.D. degree in urban development from the Technical University of Munich in 2017. She has published multiple journal articles in the research fields of transport geography, livable cities, and habitat planning.

Jinhua Zhao is currently an Associate Professor in city and transportation planning with MIT. He brings behavioral science and transportation technology together to shape travel behavior, design mobility systems, and reform urban policies. He develops methods to sense, predict, nudge, and regulate travel behavior and designs multimodal mobility systems that integrate automated and shared mobility with public transport. He directs the JTL Urban Mobility Laboratory and Transit Laboratory, MIT, and leads long-term research collaborations with major transportation authorities and operators worldwide, including London, Chicago, Hong Kong, and Singapore. He is also the Codirector of the Mobility Systems Center, MIT Energy Initiative, and the Director of the MIT Mobility Initiative.