Commun Transp Res

Research Article

Toward zero-forget continual learning for interactive trajectory prediction: A dynamically expandable approach

Huiqian Li[1], Xiaozhou Wu[1], Jin Huang[1,✉], Zhihua Zhong[1,2]

Cite this article: Li H Q, Wu X Z, Huang J, et al. *Commun Transp Res* 2026, **6**(1): 9640015. https://doi.org/10.26599/COMMTR.2026.9640015

ABSTRACT: Accurate modeling and prediction of driving behavior are crucial for enabling autonomous vehicles to safely navigate complex, interactive traffic environments. While recent continual learning approaches for interactive trajectory prediction aim to learn efficiently from streaming data, they often fail to fully retain previously learned cases when acquiring new knowledge, a phenomenon we term case-level forgetting. This limitation poses significant risks in safety-critical autonomous driving applications. This study identifies, analyzes, and addresses case-level forgetting in continual learning for trajectory prediction. We propose the dynamically expandable interactive trajectory predictor (DEITP), a novel framework that preserves previously learned knowledge through a dynamic model expansion mechanism. The mechanism regulates expansion timing by assessing model similarity, thereby controlling model growthwhile preventing catastrophic forgetting. Furthermore, to operate in realistic task-free settings where task identity is unavailable at test time, we introduce a task identification strategy based on a familiarity autoencoder that selects the most appropriate expert for prediction. Extensive experiments on real-world datasets demonstrate that DEITP substantially mitigates forgetting and achieves zero-forgetting performance when task identities are known.

KEYWORDS: continual learning; autonomous driving; dynamic expansion model; interactive behavior; catastrophic forgetting

1 Introduction

Accurately modeling and predicting the interactive behaviors of traffic participants is crucial for autonomous vehicles (AVs) to make safe and efficient decisions in real-world traffic scenarios (Gao et al, 2025). The behaviors of these participants, influenced by diverse factors in complex environments, present considerable challenges for prediction (Zhang et al., 2024). Recent studies have concentrated on collecting real-world traffic datasets that capture interactive behaviors, achieving substantial progress in behavior prediction through learning-based approaches (Ettinger et al., 2021; Li et al., 2024; Zhan et al., 2019). With the rapid development of autonomous driving technology, the volume of data being generated is expanding swiftly. AVs are expected to incrementally learn from these data, thereby continuously enhancing their performance (Cao et al., 2023; Ma et al., 2021). The technique of continual learning (CL), also known as incremental learning or lifelong learning, offers a promising means for AVs to adapt and learn throughout their operation lifespan (Wang et al., 2023).

A major challenge of continual learning is overcoming catastrophic forgetting within neural networks (Dohare et al., 2024; McCloskey and Cohen, 1989). This issue is predominantly observed when neural networks are sequentially trained across multiple tasks. Adapting to a new task's distribution often drastically diminishes performance on tasks previously mastered (Wang et al., 2024). Such degradation occurs because parameter updates, while learning a new distribution, might overwrite or disrupt previously acquired knowledge. The scenarios exhibit distinct distributions, sourced from various locations or featuring different road geometries (Zhan et al., 2019). Naive fine-tuning of neural network models with new datasets can severely impair performance in earlier learned scenarios. For instance, a model trained on datasets from roundabout scenarios may lose this specific knowledge after training on intersection scenarios (Ma et al., 2021). Consequently, this may cause inaccurate predictions in scenarios the model previously handled well. This potentially causes AVs to make unsafe decisions, whether overly cautious or perilous. Thus, maintaining previously learned knowledge while acquiring new data is vital for the safety of AVs.

Continual learning has attracted significant attention across a variety of fields (Wang et al., 2024). Initially, focused on simpler, lower-dimensional tasks such as image classification (Rebuffi et al., 2017) and reinforcement learning (Rusu et al., 2016), research has evolved to address more complex, higher-dimensional challenges, including large language models (Wu et al., 2024) and autonomous driving (Gong et al., 2024). Beyond merely addressing catastrophic forgetting, continual learning technologies strive to balance learning plasticity with memory stability while maintaining resource efficiency. Recent studies have delved into the domain of continual learning for interactive trajectory prediction, aiming to reduce the loss of previously learned knowledge upon encountering new datasets, and have achieved

[1] School of Vehicle and Mobility, Tsinghua University, Beijing 100084, China. [2] Chinese Academy of Engineering, Beijing 100088, China.

✉ Corresponding author. E-mail: huangjin@tsinghua.edu.cn

Received: October 13, 2025; Revised: December 9, 2025; Accepted: February 4, 2026

significant breakthroughs (Bao et al., 2023; Li et al., 2023; Lin et al., 2023a). However, these studies often assess method effectiveness by overall dataset performance, overlooking case-level forgetting. As shown in Fig. 1, despite favorable results at the dataset level, significant forgetting persists at the case level, potentially leading to catastrophic consequences in safety-critical applications such as autonomous driving (Zhang et al., 2023). Here, we define a "case-level forgetting event" to have occurred when the performance of an individual previous test case deteriorates beyond a certain threshold over the course of learning (Toneva et al., 2019). Although some previous studies have focused on the case-level forgetting problem in continual learning (Benkő, 2024), this issue has not received sufficient attention in the context of vehicle interaction trajectory prediction.

We concentrate on the sequential learning setting for interactive trajectory prediction and first study the case-level forgetting problem in this setting, which is crucial for autonomous driving safety. Instead of enhancing the prediction accuracy on a single dataset, we aim to develop a continual learning approach that can prevent catastrophic forgetting and achieve zero-forget on previously learned scenarios. To this end, this study introduces an architecture-based continual learning approach for interactive trajectory prediction in dynamic environments. Our contributions are outlined as follows:

- We propose the first architecture-based continual learning approach for interactive trajectory prediction, named dynamically expandable interactive trajectory predictor (DEITP). A dynamic expansion mechanism is designed to retain previous knowledge and determine the timing of expansion to balance model size and prediction performance.
- We present a solution for task identification and expert selection without task labels. A simple and effective task identification strategy based on autoencoders is developed to select the appropriate expert model during testing, allowing the proposed method to be applicable in settings where task identities are not available.
- We conduct intensive experiments to validate the proposed method using continuous real-world traffic datasets, showing its capability to effectively prevent catastrophic forgetting, achieving zero forgetting when task identities are present during testing. The code is available at https://github.com/lhquixotic/DEITP.

The structure of the remainder of this paper is as follows: Section 2 reviews related works on interactive trajectory prediction and continual learning. Section 3 formulates the vehicular interactive trajectory prediction and continual learning problem. Section 4 introduces the details of the proposed DEITP approach. The experimental setups and results are described and discussed in Section 5. Section 6 concludes the study.

2 Related work

2.1 Interactive behavior modeling and prediction

Significant research efforts have been devoted to modeling and predicting the interactive behaviors of traffic participants. Wang et al. (2022) systematically reviewed social interactions in traffic scenes and various modeling approaches. Several effective methods for interactive trajectory prediction have been proposed, including physics-based, classic machine-learning, deep learning, and reinforcement learning techniques (Huang et al., 2022).

Recent research focuses on models with stronger representation capacity and richer contextual inputs (Geng et al., 2023; Lv et al., 2024). Recent literature has focused on leveraging high-dimensional sensor data and bird's eye views (BEVs) for capturing and predicting interactions (Singh, 2023). Casas et al. (2024) proposed DeTra, a unified framework that reframes the combined task as a trajectory refinement problem. Their approach uses a refinement transformer to jointly infer object presence, pose, and multimodal future trajectories directly from LiDAR and HD maps, eliminating intermediate bottlenecks. Lang et al. (2024) developed a center-based transformer network that uses visual and contextual information from multiview images and HD semantic maps to enhance prediction accuracy. While these methods can address cumulative perception errors, they may pose challenges in development and convergence (Singh, 2023). Given their inherent capability to model complex interaction behaviors and uncertainties, graph neural networks (GNNs) have been widely adopted for trajectory prediction tasks (Jin et al., 2024; Ma et al., 2021; Wang et al., 2022). Li et al. (2021) model the interactions among traffic participants as spatial–temporal graphs and predict the trajectories of vehicles and pedestrians by a GNN-based multitask learning approach. Lu et al. (2022) designed a novel heterogeneous context-aware graph convolutional network that can simultaneously extract hidden contexts from historical trajectories, driving scenes, and intervehicle interaction behaviors. Most existing research still concentrates on improving prediction accuracy on given datasets, where training is conducted on static datasets and the closed-world assumption is followed during testing (Parmar et al., 2023). In the continual learning setting, additional measures are essential to mitigate catastrophic forgetting.

2.2 Continual learning

Continual learning methods have experienced substantial advancements in recent years (Parisi et al., 2019; Wang et al., 2025). Wang et al. (2024) review the state-of-the-art continual learning methods and divide them into five categories. Regularization-based approaches add explicit regularization terms to balance the previous and current tasks, including weight regularization and function regularization. Weight regularization

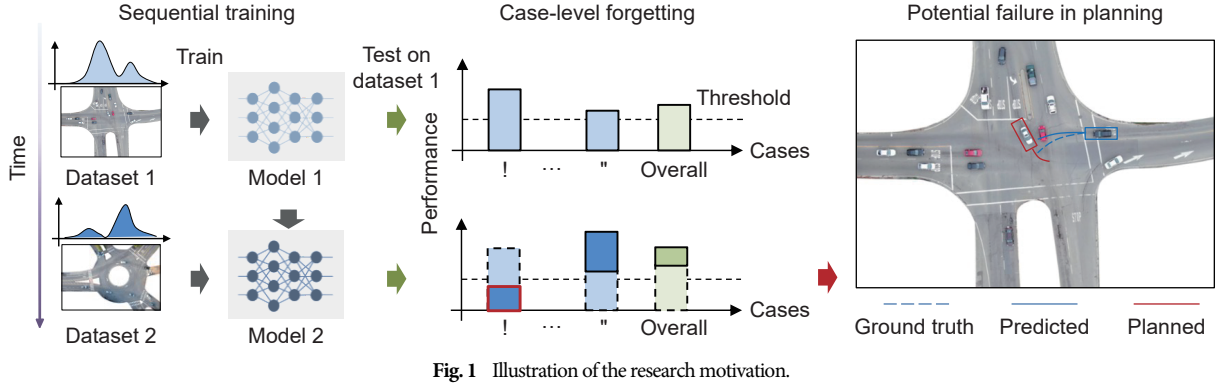


Fig. 1 Illustration of the research motivation.

selectively regularizes the variation in network parameters, such as elastic weight consolidation (EWC) (Kirkpatrick et al., 2017) and synaptic intelligence (SI) (Zenke et al., 2017). Function regularization typically employs the previously learned model as the teacher of the currently trained model by means of knowledge distillation (Gou et al., 2021) to mitigate catastrophic forgetting. Replay-based approaches approximate and recover old data distributions when training on new tasks. According to the replay content, these approaches can be classified as experience replay (ER) (Wang et al., 2021) and generative replay (GR) (van de Ven et al., 2020). Optimization-based approaches design explicit optimization programs through gradient projections (Lopez-Paz and Ranzato, 2017), loss landscapes (Davari et al., 2022) or meta-learning (Javed and White, 2019). Representation-based approaches create and exploit the strengths of representations for continual learning, such as self-supervised learning (Madaan et al., 2022) and large-scale pretraining (Mehta et al., 2023). Architecture-based methods construct task-specific parameters to resolve the catastrophic forgetting problem. Some of these methods isolate the parameters for different tasks (Serra et al., 2018), while others use a dynamic architecture to adapt to new tasks (Ramesh and Chaudhari, 2022). Despite significant progress in continual learning methods, most existing research is validated on relatively simple datasets such as the variants of MNIST, CIFAR and ImageNet. Their applicability in high-dimensional autonomous driving tasks requires further verification.

Some studies have focused on investigating the continual learning problem in interactive behavior prediction (Li et al., 2023). The study of Ma et al. (2021) is a pioneer in continual learning interactive behavior prediction using generative replay methods, mitigating the catastrophic forgetting issue. Bao et al. (2023) propose conditional Kullback–Leibler divergence to evaluate the spatiotemporal dependency difference among varied driving scenarios and develop a lifelong trajectory framework based on generative replay. Lin et al. (2023b) utilize the measurement of traffic divergence between scenarios to dynamically allocate memory and introduce a regularization-based method based on gradient episodic memory (GEM) to mitigate the catastrophic forgetting issue. Existing studies have revealed the impact of learning sequential datasets with dynamic distributions on the performance of trajectory predictors and proposed optimization-based and replay-based continual learning methods to effectively mitigate this issue. However, these methods only evaluate performance at the dataset level, ignoring case-level forgetting. In other words, while these methods prevent performance degradation at the dataset level, the prediction accuracy of some test cases may still decrease. In autonomous driving scenarios, such prediction failures can lead to catastrophic consequences. The architecture-based methods hold promise in overcoming case-level forgetting by preserving parameters, which inspires us to employ them to achieve zero-forget in the safety-critical vehicle trajectory prediction task.

3 Problem formulation

This study aims to develop a multivehicle interactive trajectory predictor trained on a stream of datasets characterized by dynamic data distributions. We first formulate the interactive trajectory prediction problem and then introduce the continual learning problem for interactive trajectory prediction.

3.1 Vehicle interactive trajectory prediction

The interactive trajectory predictor observes historical trajectories of n vehicles in the scenarios for a duration of T_{h} and predicts the future trajectories in the next T_{f} time steps. Similar to previous work (Deo and Trivedi, 2018; Li et al., 2022), the historical trajectories can be described as

$$\boldsymbol{X} = \left[\boldsymbol{q}^{(t-T_{\mathrm{h}})}, \ldots, \boldsymbol{q}^{(t-1)}, \boldsymbol{q}^{(t)}\right] \tag{1}$$

where $\boldsymbol{q}^{(t)} = [x_0^{(t)}, y_0^{(t)}, x_1^{(t)}, y_1^{(t)}, \ldots, x_n^{(t)}, y_n^{(t)}]$ are the x and y coordinates of the n vehicles in the scenario at time step t. The subscript $i = 0$ denotes the predicted vehicle, while the other subscripts $i = 1, \ldots, n$ represent the surrounding vehicles. The predicted trajectory follows a probabilistic distribution over:

$$\boldsymbol{Y} = \left[\boldsymbol{q}_0^{(t+1)}, \ldots, \boldsymbol{q}_0^{(t+T_{\mathrm{f}})}\right] \tag{2}$$

where $\boldsymbol{q}_0^{(t)} = [x_0^{(t)}, y_0^{(t)}]$ are the future coordinates of the vehicle being predicted. We assume that the future coordinates follow a bivariate Gaussian distribution such that $\boldsymbol{q}_0^{(\tau)} \sim \mathcal{N}(\boldsymbol{\mu}^{(\tau)}, \boldsymbol{\sigma}^{(\tau)}, \rho^{(\tau)})$. We denote the predicted trajectory as $\widehat{\boldsymbol{q}}_0^{(\tau)}$, which follows the estimated bivariate distribution $\mathcal{N}(\widehat{\boldsymbol{\mu}}^{(\tau)}, \widehat{\boldsymbol{\sigma}}^{(\tau)}, \widehat{\rho}^{(\tau)})$. The objective of the predictor is to minimize the negative log-likelihood, which is defined as

$$\mathcal{L}_p(\theta) = -\sum_{\tau=t+1}^{t+T_{\mathrm{f}}} \log\left(\mathrm{P}\left(\boldsymbol{q}_0^{(\tau)} | \widehat{\boldsymbol{\mu}}^{(\tau)}, \widehat{\boldsymbol{\sigma}}^{(\tau)}, \widehat{\rho}^{(\tau)}\right)\right) \tag{3}$$

where θ includes all the trainable parameters of the predictor, $\boldsymbol{\mu}^{(\tau)}$ is the mean of the distribution, $\boldsymbol{\sigma}^{(\tau)}$ is the variance, and $\rho^{(\tau)}$ is the correlation.

3.2 Continual learning for trajectory prediction

Continual learning is defined as the ability to learn from a stream of nonindependent identically distributed data aimed at preserving and enhancing previously acquired knowledge. In practice, training samples arrive in sequence, and the CL models need to learn corresponding tasks with no or limited access to previous training samples and perform well on test sets of both current and old tasks (Gunasekara et al., 2023). Formally, we define a dataset stream as $\mathcal{D}_b$, where $b \in \mathcal{B} = \{1, \ldots, k\}$ is the task identity. For the interactive trajectory prediction problem, a "task" represents training on samples $\mathcal{D}_b$ following the distribution $\mathcal{D}_b := \mathrm{P}(\boldsymbol{x}_b, \boldsymbol{y}_b)$, where $\boldsymbol{x}_b$ and $\boldsymbol{y}_b$ are the input and output sequences of the b-th task, respectively. The objective of the task is to continually learn a set of parameters θ from the current data stream without interfering with the performance of samples from previous tasks. Furthermore, based on the division of incremental batches and the availability of task identities, the continual learning problem can be categorized into eight distinct scenarios (Wang et al., 2024). In this study, we mainly regard trajectory prediction continual learning as a task-incremental learning problem, where task identities are provided in both the training and testing phases. The scenario is practical in real-world applications, since different trajectory datasets can be distinguished by the road geometry or other labels that can be provided in testing. We also consider the task-free continual learning problem in this study, where task identities are not available in either the training or testing phase.

4 Methodology

In this section, we introduce the proposed dynamically expandable interactive trajectory predictor. We first introduce the overall framework to illustrate how our method works, and then we explain the implementation of each module in detail.

4.1 Overall framework

The overall framework of the proposed DEITP is illustrated in Fig. 2. The model comprises multiple base models, each

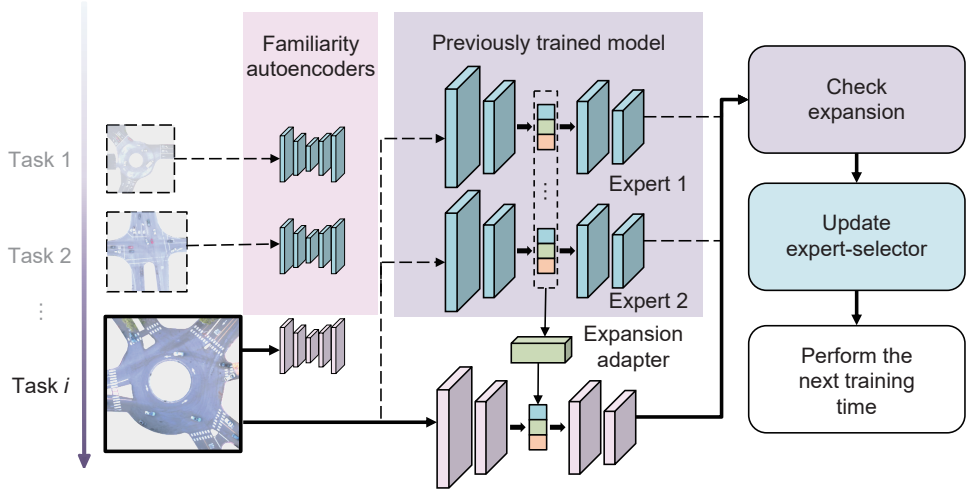


Fig. 2 Overall framework of the DEITP approach.

designated as an "expert". When a new task arrives, we freeze all existing experts to avoid interference and initialize a new expert to learn from the current task's dataset while leveraging the knowledge accumulated by previous experts. After training completion, we evaluate the new expert's knowledge base relative to that of preceding experts by measuring knowledge similarity. If the new expert has acquired distinctly different knowledge, it is retained; otherwise, it is not expanded to maintain the model size within manageable limits. The method of selectively expanding the model is referred to as "dynamically expandable". That is, we achieve "stability-plasticity" balance at the cost of extra parameter storage while trimming plasticity to some extent to avoid excessive memory usage. Next, we design a method to choose the appropriate expert for prediction during testing. If the task identities are provided in both the training and testing phases, we need to simply update the task-expert mapping, called the expert selector. Moreover, for task-free continual learning settings, we train an autoencoder to reconstruct the dataset just observed, which is used to identify which task the test case belongs to and match it with an appropriate expert for prediction. The trained autoencoders are referred to as "familiarity autoencoders (FAEs)" because their reconstruction errors across different datasets can characterize the degree of familiarity. The pseudocode used for training DEITP is described in Algorithm 1.

We focus on three key issues: (1) how to expand the model, (2) when to expand it, and (3) which expert to select during testing. These issues will be discussed thoroughly in the following subsections.

Algorithm 1 Training algorithm for DEITP

Input: Dataset stream $\mathcal{D}_b, b \in \{1, \ldots, n\}$; base model $\mathcal{M}_\theta$; lateral adapter h_U; expert set $\mathbf{Q} = \{\}$; expert number $k = 0$; expansion threshold β; expert selector $\mathcal{T}$; familiarity autoencoder $\mathcal{G}$.

Output: Expert set $\mathbf{Q} = \{Q_1, \ldots, Q_k\}$.

1: Learning from the first dataset:

2: Initialize the first expert Q_1 with $\mathcal{M}_\theta$.

3: Train the first expert Q_1 on D_1.

4: $\mathbf{Q} \leftarrow \{Q_1\}, k \leftarrow 1, \mathcal{T}(b) \leftarrow 1$.

5: Learning from the sequential datasets:

6: **for** dataset $b = 2$ to n **do**

7: Freeze the previous experts $Q_{1:k}$.

8: Initialize a new expert Q_{k+1} with $\mathcal{M}_\theta$.

9: Add the lateral adapter h_{U_k} from Q_k to Q_{k+1}.

10: Train the current expert Q_{k+1} on $\mathcal{D}_b$.

11: $R_1^{(b)}, \ldots, R_{k+1}^{(b)} \leftarrow$ Test $Q_1, \ldots, Q_{k+1}$ on $\mathcal{D}_b$.

12: **if** $\min\{R_1^{(b)}, \ldots, R_k^{(b)}\} - R_{k+1}^{(b)} \leq \beta$ **then**

13: $\mathcal{T}(b) \leftarrow \text{argmin}_{i=1}^{k}\{R_i^{(b)}\}$.

14: **else**

15: $\mathbf{Q} \leftarrow \mathbf{Q} \cup \{Q_{k+1}\}, k \leftarrow k + 1$.

16: $\mathcal{T}(b) \leftarrow k$;

17: **end if**

18: **end for**

4.2 Model expansion method

To avoid forgetting previous knowledge, we propose an architecture-based continual learning approach such as a progressive neural network (PNN) (Rusu et al., 2016). The method is characterized by utilizing new neurons to learn new knowledge, thereby avoiding interference with previous knowledge. Unlike dynamically expandable networks (DENs) (Yoon et al., 2018), we keep the expansion structure identical to the base model, enabling the new model to learn more effectively. Let $\mathbf{Q} = \{\mathcal{Q}_1, \ldots, \mathcal{Q}_k\}$ be the DEITP model with k experts. For the first task, we train an expert $\mathcal{Q}_1$ with the same architecture as the base prediction model $\mathcal{M}_\theta$. When learning from a new task, we initialize a new expert $\mathcal{Q}_j$ consisting of a predictor $\mathcal{M}_{\theta_j}$ and lateral connections h_{U_j}. Meanwhile, we freeze the parameters of the previously learned experts, which means stopping updating the parameters during training. The lateral connection h_{U_j} is designed to leverage the knowledge gained from previous tasks. It creates interexpert connections by routing feature maps from the preceding layers of existing experts to the input layers of newly added experts, mathematically represented as

$$o_j^{(m)} = \sigma\left(W_j^{(m)} o_j^{(m-1)} + \sum_{l>1}^{j} U_l^{(m)} o_{l-1}^{(m-1)}\right) \tag{4}$$

where $o_j^{(m)}$ is the output of the m-th layer of the j-th expert, $W_j^{(m)}$ is the weight of the m-th layer of the j-th expert, and $U_l^{(m)}$ is the weight of the adapter network between l-th and $(l+1)$-th experts. σ represents the activation function. The previous expert can be selected as the expert with the best performance on the

training dataset of the new task. This design is helpful to accelerate the convergence of the new expert. The number and architecture of lateral connections are determined by the architecture of the original architecture of the predictor, referred to as the "base model". In this study, we choose Social Spatio-Temporal Graph Convolutional Neural Network (Social-STGCNN) as the base model (Mohamed et al., 2020), while other base models can also be used. Social-STGCNN is a graph-based interaction-aware trajectory prediction model. It takes graph representations of vehicles' positions as input into the STGCNNs, followed by time-extrapolator convolutional neural networks (TXPCNNs) to output the predicted distribution of trajectories. In this case, lateral connection modules are added between two adjacent layers. The module consists of a convolutional layer and an activation layer. A visualized expansion example is illustrated in Fig. 3. After expansion, the model is trained on the new task data until the loss converges. A new trained expert model is obtained, which can perform well on the new task.

4.3 Dynamical expansion mechanism

Expansion may not be necessary if the existing network sufficiently addresses the new task (Yoon et al., 2018). Dynamically expanding the model saves storage and computational consumption. Prior research typically identifies outlier samples (Lee et al., 2020; Rao et al., 2019) or measures the novelty of the knowledge (Ye and Bors, 2023) to trigger model expansion. In this section, we propose a dynamic expansion mechanism that evaluates the similarity between the newly trained model and previous experts and initiates expansion when the new expert significantly diverges from them. Thus, the expansion signal can be denoted as

$$\mathcal{L}_s(\mathcal{Q}_1, \mathcal{Q}_k), \ldots, \mathcal{L}_s(\mathcal{Q}_{k-1}, \mathcal{Q}_k)\} > \beta \tag{5}$$

where $\mathcal{L}_s(\mathcal{Q}_j, \mathcal{Q}_k)$ is the similarity measure function, which is designed as

$$\mathcal{L}_s(\mathcal{Q}_j, \mathcal{Q}_k) = \mathbb{E}_{\mathrm{X}\sim\mathcal{D}_k}\left[||\mathcal{Q}_j(\mathrm{X}) - \mathcal{Q}_k(\mathrm{X})||_2\right] \tag{6}$$

where $\mathcal{Q}_k(\mathrm{X})$ is the prediction result of expert k-th, and $\mathcal{D}_k$ is the dataset that expert k-th is trained on. The smaller this value is, the more similar the two experts are. For simplicity, we use the following calculation method in practical implementation:

$$\mathcal{L}_s(\mathcal{Q}_j, \mathcal{Q}_k) = R_j^{(i)} - R_k^{(i)} \tag{7}$$

where $R_k^{(i)}$ is the prediction accuracy metric of test performance, as detailed in Section 5.3, of the k-th expert on the i-th task. We assess the similarity of the experts based on their test performance, a straightforward yet effective approach to ensure that model expansion contributes significantly to performance enhancement. If two experts, regardless of their internal differences, yield comparable predictive performance, our framework treats them as functionally equivalent and refrains from introducing redundancy. This method also avoids additional data storage for similarity evaluation. We dynamically add a new expert $\mathcal{Q}_k$ to $\mathbf{Q}$ if the condition specified in Eq. (5) is satisfied after training. The expansion threshold, β, plays a crucial role in balancing model size and performance. Setting a lower β promotes model expansion, facilitating performance improvements at the potential cost of increased model size. Conversely, a higher β supports maintaining a smaller, more compact model, potentially at the expense of maximal performance.

4.4 Expert selector and task identification

To select an appropriate expert during testing, we need a mapping from the test input to the expert index, which is called the expert selector. When task identities are known for both training and testing cases, the expert selector can be straightforwardly represented as a function mapping task identity to the expert index, denoted as $\mathcal{T}: \mathbb{R} \to \mathbb{R}$. However, for the task-free continual learning problem, the task identity of the test cases remains unknown, so it becomes essential to detect the identity or determine the appropriate expert for each input (Chen et al., 2025). Autoencoder architectures are known for their effectiveness in reconstructing data for applications such as novelty detection (Jacobson et al., 2022) and driving scenario recognition (Deng et al., 2021). In this work, we introduce a scenario-aware familiarity autoencoder designed to identify the scenario of the test case. Specifically, for each task dataset, we train not only an interactive trajectory predictor but also a familiarity autoencoder. This autoencoder is tasked with reconstructing the data corresponding to its designated task, thereby enabling accurate expert selection during testing. During testing, we reconstruct the case using all FAEs and select the scenario corresponding to the FAE with the minimum reconstruction error as the scenario to which the case belongs, which can be denoted as

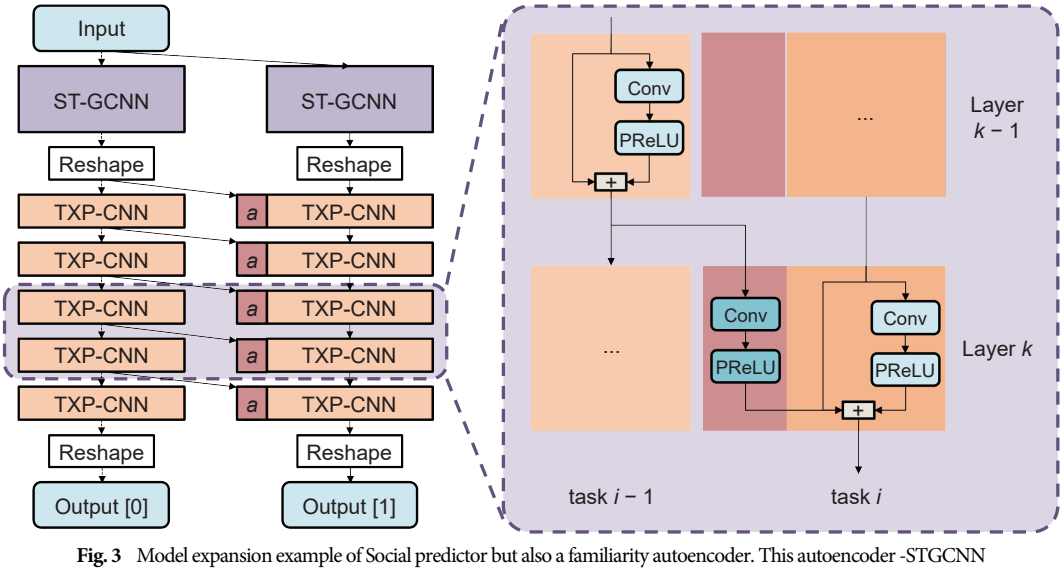


Fig. 3 Model expansion example of Social predictor but also a familiarity autoencoder. This autoencoder -STGCNN

$$\widehat{b} = \operatorname{argmin}_{b\in\{1,\ldots,n\}} \mathcal{L}_{\mathrm{r}}\left(\mathcal{G}_b\left(\mathbf{X}\right), \mathbf{X}\right) \tag{8}$$

where $\widehat{b}$ is the predicted scenario index, $\mathbf{X}$ is the test case input, $\mathcal{G}_b(\mathbf{X})$ is the reconstructed input, and $\mathcal{L}_{\mathrm{r}}$ is the reconstruction loss function. The choice of function depends on the architecture of the FAE, which can be the mean squared error (MSE) or the Kullback-Leibler divergence. In this study, we use the spatial-temporal graph autoencoder to reconstruct the scenario distribution.

5 Experiments

To investigate the performance of the proposed dynamic expansion model, we conduct continual learning experiments using the INTERACTION datasets (Zhan et al., 2019). We evaluate the prediction accuracy performance and the ability to overcome catastrophic forgetting compared with several baseline methods. Moreover, we compare the storage usage required by different methods.

5.1 Datasets

We utilize the INTERACTION dataset for validation because it contains highly interactive, multivehicle trajectory data across a wide variety of road geometries and traffic scenarios. Its nonstationary and distinctly shifting data distributions make it an ideal benchmark for evaluating continual learning methods under distribution shifts. Specifically, we select five different scenarios in the experiments, marked as MA, FT, ZS, EP and SR, each of which represents the geographic location where the data are collected. The MA scenario corresponds to an urban intersection, ZS represents a ramp merging situation, and FT, EP and SR are three different urban roundabouts. Moreover, we divide each scenario dataset into subsets as tasks at different times. For each task, we further divide the dataset into training, validation, and testing sets at a ratio of 7:1:2. The target of each task is to observe 2-s historical trajectories and predict the future 4-s trajectories. The original data provided by INTERACTION include the IDs and coordinates of vehicles with timestamps. We preprocess the datasets following previous work (Lin et al., 2023b). We extract the vehicle trajectories with interactions and split them into short sequences. Then, they are converted into graph-based representations, where nodes represent vehicle positions and edges with a weighted matrix encode the spatial interdependency between vehicles. The detailed scripts for data processing can be found in our code repository.

5.2 Experimental settings

The target of the experiments is to evaluate different methods on sequential learning tasks. We split each dataset into two parts of equal size and arrange them in the order "MA-FT-ZS-EP-SR" to form a task stream containing 10 tasks. The model cannot access the previous task data during training. We use the Social-STGCNN as the base predictor, which is a graph-based model. The key hyperparameters of the model can be seen in Table 1. For each task, the model is trained for up to 250 epochs, and we use an early stop strategy to stop training if the validation loss does not decrease for 50 epochs. FAE is implemented with encoders and decoders consisting of two STGCNNs.

We compare our proposed methods with several baselines as follows:

- **Joint training**. A noncontinuous learning approach that trains a model on combined datasets from all scenarios. It is regarded as the best possible performance over any continual learning method.

Table 1 Important hyperparameters

Parameter	Value
Observation horizon	2
Prediction horizon	4
Input size	2
Output size	5
STGCNN layers number	1
TXPCNN layers number	5
Learning rate	0.01
Epochs number	250
Kernel size	3
Expansion threshold	0.1
Forgetting threshold	1.5
FAE encoder layers number	2
FAE decoder layers number	2
FAE latent size	32

- **Fine tuning**. Fine tuning the base model on the new task without applying any continual learning approach.
- **Multiple models** (MM). Training multiple base models, one for each task, without any knowledge transfer.
- **EWC** (Kirkpatrick et al., 2017). Addressing catastrophic forgetting by adding a regularization term to the loss function during training. This term penalizes changes to the most important weights for previously learned tasks. The importance of each weight is determined by how much it contributes to the performance on the old tasks, measured through the Fisher Information Matrix.
- **GSM** (Lin et al., 2023b). An implementation of GEM in continuous traffic scenarios. The GSM calculates the gradients for the new scenario and compares them against the gradients for the scenarios stored in the memory. If the new gradients increase the loss in any previous scenarios, GSM projects the gradients onto a constraint that ensures no increase in the loss for the remembered scenarios.
- **PNN** (Rusu et al., 2016). Using lateral connections from networks learned by previous tasks to the current task. These connections allow features learned by previous networks to be reused by the current network, which can help in learning new tasks faster and more effectively without forgetting old tasks.
- **DEITP (ours)**. Expanding the model dynamically when knowledge learned from the previous cannot handle the new task well.
- **DEITP-FAE (ours)**. DEITP model with familiarity autoencoders to recognize the scenarios during testing in a task-free continual learning setting.

5.3 Metrics

We evaluate the performance of continual learning approaches from four aspects: prediction accuracy, forgetting measure, storage usage and inference time.

- **Prediction accuracy**. The average displacement error (ADE) and final displacement error (FDE) are two metrics commonly used in trajectory prediction (Mohamed et al., 2020). For each test case, we sample the predicted trajectory five times and compute the average of the metrics. ADE is denoted as the average Euclidean distance of the predicted coordinates and the ground truth coordinates at each time step in the prediction horizon, formulated as

$$\mathrm{ADE} = \frac{\sum_{i=1}^{N}\sum_{t'=t+1}^{t+T_f} \|\widehat{\boldsymbol{q}}_{0,i}^{(t')} - \boldsymbol{q}_{0,i}^{(t')}\|_2}{NT_f} \tag{9}$$

where N is the sample number of the test set and $\widehat{\boldsymbol{q}}_{0,i}^{(t)}$ and $\boldsymbol{q}_{0,i}^{(t)}$ denote the i-th sample of predicted and ground truth coordinates of the target vehicle at time t. Similarly, FDE represents the Euclidean distance between the final predicted position and the ground truth, written as

$$\mathrm{FDE} = \frac{\sum_{i=1}^{N} \|\widehat{\boldsymbol{q}}_{0,i}^{(t+T_f)} - \boldsymbol{q}_{0,i}^{(t+T_f)}\|_2}{N} \tag{10}$$

We use the average error (AE) to measure the performance across all training datasets, which is defined as

$$\mathrm{AE} = \frac{1}{M(M+1)/2}\sum_{j\le k}^{M} R_{k,j} \tag{11}$$

where M is the number of training datasets and $R_{k,j}$ is the test error (ADE/FDE) at the j-th task after training on the k-th task.

- **Forgetting measure.** We evaluate the performance of overcoming catastrophic forgetting from both the dataset level and case level. First, we adopt the average forgetting (AF) metric to measure the average decrease in performance on the previous tasks after observing the new ones, which is defined as

$$\mathrm{AF} = \frac{1}{M(M-1)/2}\sum_{k=2}^{M}\sum_{j<k}^{M} R_{k,j} - R_{j,j} \tag{12}$$

The degree of catastrophic forgetting can be represented by AF. The larger the value of AF is, the greater the degree of forgetting (Ma et al., 2021). Second, we define the average forgetting ratio (AFR) to measure the ratio of forgetting occurring at the case level, which is defined as

$$\frac{1}{M(M-1)/2}\sum_{k=2}^{M}\sum_{j<k}^{M}\frac{1}{N_k}\sum_{i=1}^{N_k} 1\left(r_{i,k,j} - r_{i,j,j} - \mathcal{C}\right) \tag{13}$$

where $r_{i,k,j}$ is the test error at the i-th sample of the k-th task predicted by the model after training on the j-th task. N_k is the sample number of the k-th task, and 1 is the indicator function that returns 1 if the input is larger than 0 and 0 otherwise. $\mathcal{C}$ is the forgetting threshold that determines what degree of performance degradation is considered forgetting. The larger the value of AFR is, the greater the degree of forgetting at the case level.

- **Storage usage**. We also use the storage requirement for continual learning to evaluate the performance of the methods. The storage usage includes the model parameters, training data samples, and other auxiliary data.
- **Inference time**. To assess practical usability and deployment efficiency, we measure average inference time: We randomly draw a fixed set of test samples, let the model predict on all of them, and report the mean latency. All experiments are conducted on a MacBook Pro equipped with an Apple M1 Pro chip.

5.4 Experimental results and analysis

5.4.1 Performance on all previous tasks

Initially, we assess the models' performance on prior tasks after being sequentially trained on a series of continuous datasets. Each model undergoes sequential training across 10 datasets, and following the completion of training for each dataset, it is evaluated on both the current and previous task test sets. Figs. 4a and 4b illustrate the errors across all 10 tasks upon the completion of all training sessions. The x-axis denotes the index of the test dataset, while the y-axis represents the values of ADE or FDE. It is evident that the vanilla method, which relies solely on fine-tuning for new tasks, is susceptible to catastrophic forgetting. Specifically, after training on all tasks, the vanilla method exhibits poor performance on earlier tasks with differing distributions from the final task. For example, while the continuous datasets end with SR-2, the latest model shows a marked performance decline on all previous tasks except SR-1. The EWC and GSM methods mitigate this issue, with notably less performance degradation in previous scenarios. Architecture-based continual learning methods, such as the PNN and DEITP proposed in this study, effectively maintain performance across earlier scenarios by retaining knowledge, namely, model parameters, learned from earlier tasks, closely matching the performance achieved through joint training. The measurable performance improvement over the MM baseline validates the effectiveness of the lateral connection architecture. Joint training is regarded as the optimal benchmark for evaluating continual learning approaches. Furthermore, DEITP's dynamic expansion mechanism allows it to effectively reuse knowledge from tasks MA-1 to SR-1, and it demonstrates good performance in the MA-2 to SR-2 tasks. Additionally, Table 2 offers a comparison of performance metrics for all tasks, calculated according to Eqs. (11) and (13).

5.4.2 Performance on the initial task

Next, we assess the ability of the proposed methods to prevent catastrophic forgetting. Figs. 4c and 4d illustrate the test performance on the initial task (MA-1) following training on subsequent continuous tasks. The fine-tuning method exhibits significant catastrophic forgetting, as evidenced by its poor performance on the initial tasks after learning new tasks. GSM alleviates this issue by regularizing gradients between current and previous tasks. However, in some scenarios, such as ZS-2, the level of forgetting remains pronounced. We also compute the forgetting ratio for various methods across previous tasks after training on all 10 datasets. The forgetting threshold is set as $\mathcal{C} = 1.5$ m. Any test case exhibiting an error increase exceeding this value relative to the previous model's performance is viewed as a forgetting event. Despite the effectiveness of the GSM method in reducing forgetting, it still results in more than 3% of cases being forgotten. This could lead to unexpected behaviors in autonomous vehicles during real-world operations. In contrast to GSM, PNN and DEITP effectively retain the knowledge acquired from previous tasks and consistently demonstrate robust performance on these tasks after learning new scenarios. We further visualize the proposed method's ability to overcome case-level forgetting through Fig. 5. We compare the model trained only on the MA scenario with one that has sequentially learned the MA-FT-ZS task stream and then evaluate both on the MA task. We plot the distribution of per-case performance degradation after learning new tasks. This shows that without any continual learning strategy, most cases suffer severe degradation; even with the GSM baseline, the distribution's peak is still right of zero, indicating that most cases still forget. In contrast, DEITP retains prior knowledge and overcomes forgetting; under the task-free continual learning setting, the FAE module effectively prevents case-level forgetting for most instances. On the other hand, as shown in Fig. 4, when the distribution of the new task (MA-2) is the same as that of the initial task (MA-1), DEITP cannot improve its performance on the initial task by learning from the new data, unlike the fine-tuning approach. This is a limitation of DEITP, which we will address in future work.

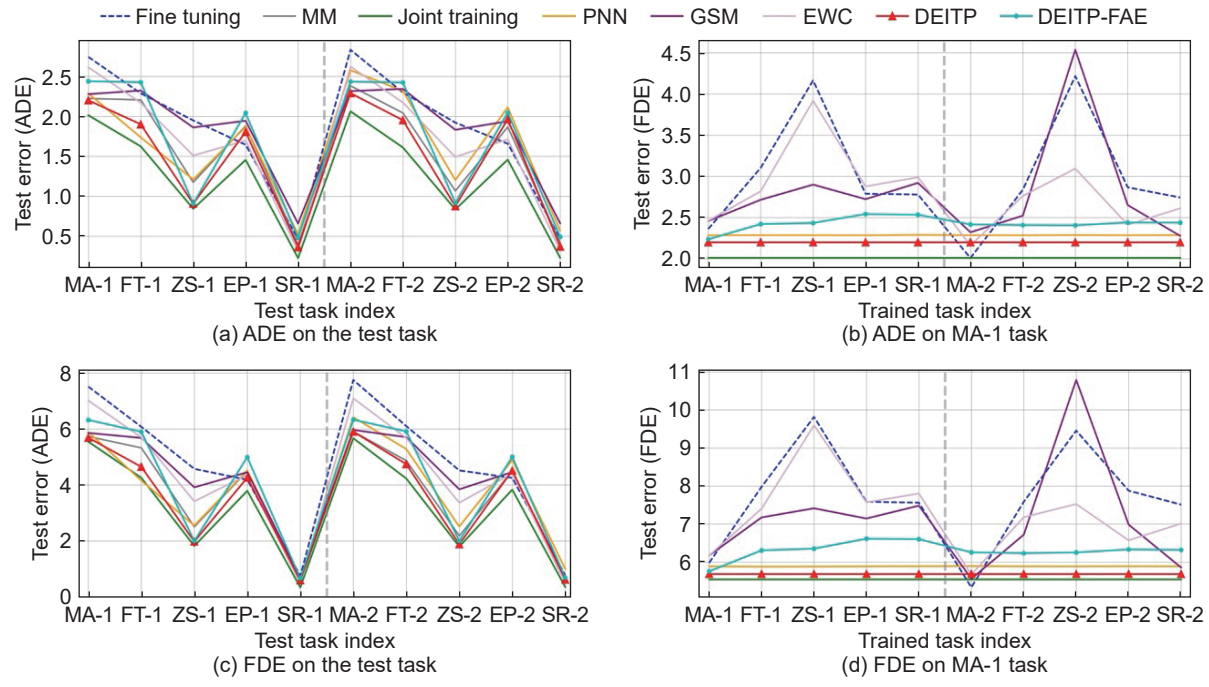


Fig. 4 Evaluation of the proposed methods after trained on continuous scenarios. (a, b) Test error (ADE and FDE) of different methods on all previous tasks after training on all 10 tasks. (c, d) Test error (ADE and FDE) of different methods on the first task (MA-1) after training on subsequent continuous tasks.

Table 2 Evaluation of the interactive trajectory prediction performance in continuous scenarios

Method	AE↓ (m)	AF↓ (m)	AFR↓	Storage↓ (MB)	Time↓ (ms)
Joint training	1.24/3.17	–/–	–/–	0.32(+ 2097*)	1.4
Fine tuning	2.19/5.34	0.81/1.94	0.22/0.50	0.32	1.4
GSM	2.11/4.91	0.48/1.14	0.12/0.32	0.32(+ 456*)	1.4
EWC	2.04/4.94	0.54/1.2	0.11/0.38	0.32(+ 456*)	1.4
PNN	1.73/4.07	0.00/0.00	0.00/0.00	6.3	17.4
DEITP	1.59/3.85	0.00/0.00	0.00/0.00	2.4	5.6
DEITP-KM	2.49/5.43	0.60/0.89	0.33/0.44	2.4	5.7
DEITP-FAE	1.79/4.37	0.11/0.29	0.03/0.08	3.0	23.5

Note: * indicates the memory space occupation caused by storing data or intermediate variables.

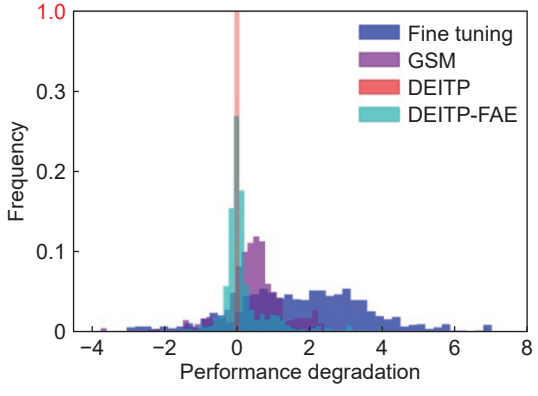


Fig. 5 Distribution of case-level degradation after learning new tasks with different methods.

5.4.3 *Performance under the task-free continual learning setting*

We evaluate our task identification method in a task-free continual learning setting. Each time a new expert is trained for a new scenario, we concurrently train and store an FAE for that scenario's input distribution. At test time, when task identity is unknown, we reconstruct the input with all stored FAEs and select the expert associated with the FAE yielding the lowest reconstruction error for trajectory prediction. We also propose a lightweight k-means clustering-based task identification method for comparison, which assigns a task label by measuring the distance between a test case and the cluster centers of each task. We refer to the methods using these two task identification approaches as DEITP-FAE and DEITP-KM, respectively.

Fig. 6 shows the expert selection accuracy of our method after learning across 10 consecutive scenarios. The heatmap displays the proportion of test cases from each scenario that are assigned to different experts. The x-axis denotes the test scenario index, and the y-axis denotes the expert index. Each cell value represents the fraction of test cases mapped to the corresponding expert, with cells matching the ground-truth expert highlighted in yellow. Expert identification accuracy varies significantly across scenarios: It reaches 98% for the ZS scenario but drops below 40% for SR. The MA, FT, and EP scenarios also exhibit notable misidentification, likely due to overlapping data distributions that make it difficult to distinguish tasks based solely on input features. To visualize this challenge, Fig. 7 presents a t-SNE embedding of case features from all scenarios. Features from the ZS scenario (red markers) form a clearly separated cluster, explaining its high identification accuracy. In contrast, features from other scenarios are tightly intermixed, leading to confusion among experts. This observation is consistent with findings in (Lin et al., 2023b), which

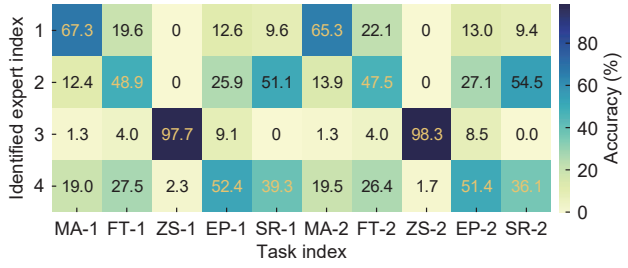


Fig. 6 Heatmap for expert identification result.

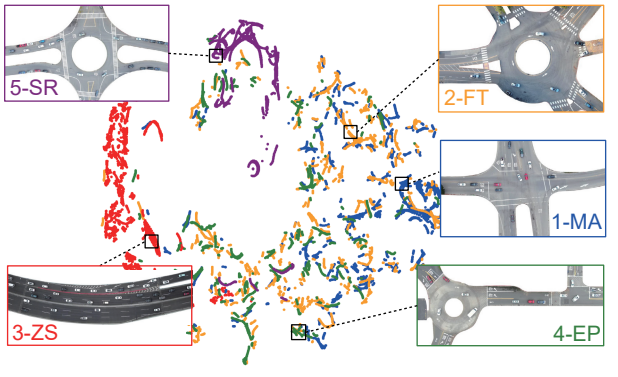


Fig. 7 t-SNE visualization of the five scenarios. Each mark represents a case of the dataset, and the color denotes the scenario index.

characterizes traffic divergence using conditional Kullback–Leibler divergence.

However, expert misidentification does not necessarily lead to poor trajectory prediction performance. Fig. 8 compares the performance of DEITP-FAE after training on 10 continuous tasks. In the ZS scenario, which achieves the highest identification accuracy, the prediction performance is almost identical to that obtained with known task identities. Interestingly, even in the SR scenario, which has the lowest identification accuracy, the prediction accuracy does not degrade significantly. This indicates that misidentified experts can still produce fairly accurate predictions for this scenario. We argue that similar interactive patterns across these scenarios lead to comparable performance among expert models trained on their respective datasets. Fig. 8 also demonstrates that the FAE-based identification method outperforms both the K-means-based approach and random selection. Expert identification errors complicate efforts to mitigate the forgetting of previous tasks. The right side of Fig. 4 shows the performance on MA-1 after sequentially learning subsequent scenarios using DEITP-FAE. Although the prediction accuracy declines slightly compared to the oracle setting with known task identities, it significantly outperforms both fine-tuning and GSM. Table 2 reports and compares the metrics for DEITP-FAE, revealing a substantial improvement over GSM and approaching near-zero forgetting. In summary, the effectiveness of our dynamically expandable approach hinges critically on the accuracy of expert identification in the task-free setting, while the success of the FAE–based method depends on how distinct the scenario-specific features are.

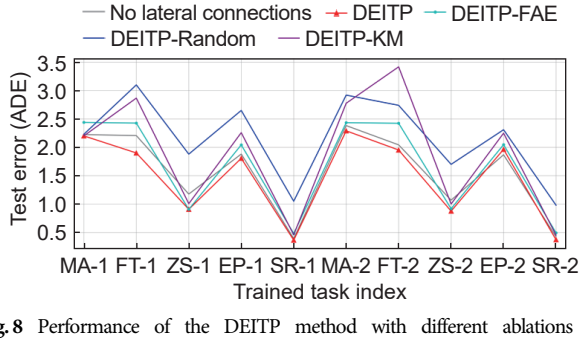


Fig. 8 Performance of the DEITP method with different ablations on 10 continuous tasks.

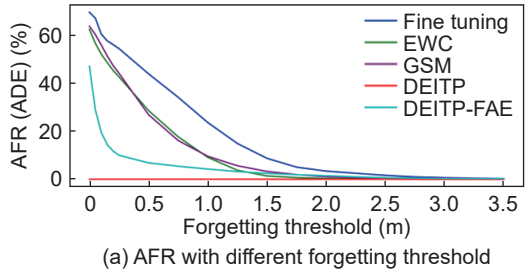
5.4.4 *Sensitivity analysis*

In this subsection, we conduct a sensitivity analysis to demonstrate the robustness of the proposed method to key hyperparameter selection and varying experimental settings.

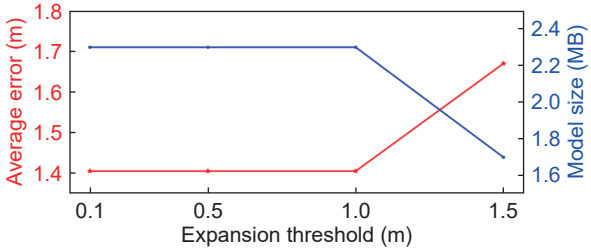
Selection of the forgetting threshold. First, to quantify the sensitivity of AFR to the selection of forgetting threshold $\mathcal{C}$, we vary the threshold from 0 to 3.5 and plot the curve of AFR in Fig. 9a. DEITP maintains zero forgetting across the whole range because it preserves previous knowledge. DEITP-FAE shows advantages over other continual learning baselines when the threshold is below 1.5; above 1.5, the margin shrinks but still outperforms naive fine-tuning. This indicates that our claimed reduction in the forgetting rate holds across multiple thresholds. We select 1.5 as the default since the value corresponds to half the width of a typical lane.

Selection of the expansion threshold. Next, we evaluate the impact of the selection of the expansion threshold β. We continuously train across five tasks using various expansion thresholds and compare the average ADE across all tasks and the final model size. The results displayed in Fig. 9b indicate that, in our experiments, the performance is not sensitive when the expansion threshold is below 1.5. However, when it becomes larger (above 1.5), the model tends to retain fewer experts, leading to a degradation in prediction performance. We set this threshold as 0.1, indicating that if the reduction in prediction error by the current expert relative to the previous expert does not surpass 0.1, then expansion is not pursued.

Different task streams. In addition, we train and test the proposed method with another dataset order (EP-FT-SR-ZS-MA). The results show that the performance of DEITP on different tasks varies slightly and can still achieve zero forgetting. This demonstrates that the proposed method is robust to different task sequences. We have also validated the effectiveness of the proposed method on the task stream defined with different traffic densities.



(a) AFR with different forgetting threshold



(b) Average error and model size with expansion threshold

Fig. 9 Sensitivity analysis for the selection of (a) forgetting threshold $\mathcal{C}$ and (b) expansion threshold β.

5.4.5 Storage usage and inference time

Finally, we compare the storage usage and inference time of various continual learning methods. In addition to the model parameters, different approaches may retain intermediate data or additional parameters. Table 2 details the storage utilized by each method. Both the joint training and fine-tuning methods do not retain any extra information, so their storage requirements are solely dependent on the size of the base model. In contrast, the GSM approach maintains its original model architecture but stores previous data to regularize gradients for new tasks. This results in storage consumption significantly exceeding the base model itself. These methods maintain the original model architecture, resulting in identical inference times to the base model. Dynamic architecture-based continual learning methods, such as PNN and DEITP, encapsulate previous knowledge within parameters. As depicted in Fig. 10, the PNN method adds a new column for each task, whereas DEITP employs a dynamic expansion mechanism to determine the necessity of expansion. For instance, the expert trained on the EP-1 scenario exhibits performance comparable to that of the SR-1 expert, thus negating the need for expansion in that instance. Similarly, for the next five datasets from MA-2 to SR-2, which share the same distribution as the learned datasets, previous experts have demonstrated accurate predictions. Therefore, DEITP requires only four experts to predict across 10 consecutive scenarios, with the final model size merely 2.4 MB, 62% smaller than PNN. Although FAE-based task identification requires additional memory, its storage demand remains minimal compared to the base model, highlighting DEITP's efficient balance between functionality and model size control. Furthermore, the approach avoids continuous increases in inference time when the model expands, making it practically scalable for real-world deployment. Table 2 shows that the inference time of the DEITP on the Apple M1 Pro chip is only 5.6 ms. Although this is higher than the method without expansion, it still meets the requirements of practical use. When FAE is used for task identification, the inference time increases to 23.5 ms. This increase is due to the computation of multiple FAEs. However, in real-world applications, we can identify task changes and select experts at a lower frequency, which will further optimize the overall inference time.

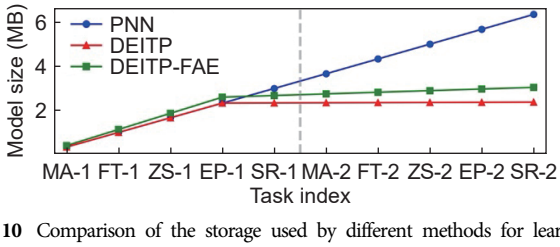


Fig. 10 Comparison of the storage used by different methods for learning 10 continuous scenarios.

6 Conclusions

This study introduces a novel architecture-based approach to interactive trajectory prediction in continual learning, termed DEITP. This method leverages a dynamic expansion mechanism that enables learning from sequentially incoming traffic scenario datasets. An expert is trained for each new task, and its performance is compared with that of existing experts. The new expert is expanded within the model if its performance improvement surpasses a predefined threshold. This strategy effectively prevents catastrophic forgetting by preserving previously acquired knowledge within the model parameters. Additionally, DEITP balances model size and predictive performance, thereby avoiding an overly large model or excessive memory consumption. Furthermore, we introduce an innovative expert identification method using a familiarity autoencoder in a task-free continual learning setting. This method identifies task identities and selects the appropriate expert for a given test input based on reconstruction errors.

We conduct experiments using 10 continuous scenarios from the INTERACTION dataset to evaluate our proposed method. The experimental results indicate that our method surpasses the baseline models in terms of prediction accuracy. When task identities are available, our approach achieves zero forgetting. In the task-free continual learning setting, our method demonstrates an average forgetting ratio of just 2% at the case level, which is significantly lower than that observed with the baseline GSM approach. Furthermore, our dynamic expansion model reduces storage usage by over 60% compared to PNNs. These findings underscore the effectiveness of our method in both preventing catastrophic forgetting and optimizing storage usage utilization.

The proposed DEITP approach significantly enhances our ability to overcome catastrophic forgetting, outperforming existing methods for interactive trajectory prediction. This versatile method works across model architectures and is especially effective in storage-limited scenarios. However, the proposed method does present some limitations in terms of performance and real-world applications. Designed for overcoming case-level forgetting, the model tends to overlook potential enhancements that new datasets might offer. In the setting of task-free continual learning, the task identification accuracy is limited by the current method. In addition, although the proposed method does not impose strict requirements on the distribution of emerging datasets in real-world applications, it necessitates relatively balanced data quantities, which may limit its applicability in few-shot learning scenarios. In future work, we will enhance the performance of existing experts in new scenarios. We also plan to use more advanced models to improve task identification accuracy.

Author contributions

Huiqian Li: Writing – original draft, Methodology, Data curation, Conceptualization, Software, Visualization, Formal analysis, and validation. **Xiaozhou Wu:** Writing – original draft, Methodology, Data curation, Validation. **Jin Huang:** Supervision, Writing – review & editing. **Zhihua Zhong:** Supervision, Writing – review & editing.

Replication and data sharing

The data and codes used in this study are available at https://doi.org/10.26599/ETSD.2026.9190004.

Acknowledgements

This work was supported in part by the National Key R&D Program of China (Nos. 2023YFF0615802 and 2024YFB2505101), the Anhui Provincial Major Science and Technology Special Project (No. 202423d120500072), and China Scholarship Council (CSC).

Declaration of competing interest

The authors have no competing interests to declare that are relevant to the content of this article. The author Jin Huang is the Editorial Committee member of this journal.

References

Bao, P., Chen, Z., Wang, J., Dai, D., Zhao, H., 2023. Lifelong vehicle tra-

jectory prediction framework based on generative replay. *IEEE Trans Intell Transp Syst*, **24**, 13729–13741.

Benkő, B., 2024. Example forgetting and rehearsal in continual learning. *Pattern Recognit Lett*, **179**, 65–72.

Cao, Z., Jiang, K., Zhou, W., Xu, S., Peng, H., Yang, D., 2023. Continuous improvement of self-driving cars using dynamic confidence-aware reinforcement learning. *Nat Mach Intell*, **5**, 145–158.

Casas, S., Agro, B., Mao, J., Gilles, T., Cui, A., Li, T., Urtasun, R., 2024. Detra: A unified model for object detection and trajectory forecasting. In: European Conference on Computer Vision, 326-342.

Chen, L., Dong, T., Li, X., Xu, X., 2025. Logistics engineering management in the platform supply chain: An overview from logistics service strategy selection perspective. *Engineering*, **47**, 236–249.

Davari, M., Asadi, N., Mudur, S., Aljundi, R., Belilovsky, E., 2022. Probing representation forgetting in supervised and unsupervised continual learning. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 16691–16700.

Deng, N., Jiang, K., Cao, Z., Zhou, W., Yang, D., 2021. Decision-Oriented Driving Scenario Recognition Based on Unsupervised Learning. In: International Conference on Transportation Professionals 2021 (CICTP), 564–573.

Deo, N., Trivedi, M. M., 2018. Convolutional social pooling for vehicle trajectory prediction. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 1549–15498.

Dohare, S., Hernandez-Garcia, J. F., Lan, Q., Rahman, P., Mahmood, A. R., Sutton, R. S., 2024. Loss of plasticity in deep continual learning. *Nature*, **632**, 768–774.

Ettinger, S., Cheng, S., Caine, B., Liu, C., Zhao, H., Pradhan, S., et al., 2021. Large scale interactive motion forecasting for autonomous driving: The waymo open motion dataset. In: 2021 IEEE/CVF International Conference on Computer Vision (ICCV), 9690–9699.

Gao, Z., Jia, B., Xie, D., Wang, W., Wu, J., 2025. A discussion on the complexity and transit mechanisms of urban traffic systems. *Engineering*, **44**, 24–29.

Geng, M., Chen, Y., Xia, Y., Chen, X. M., 2023. Dynamic-learning spatial-temporal Transformer network for vehicular trajectory prediction at urban intersections. *Transp Res Part C Emerg Technol*, **156**, 104330.

Gong, C., Lu, C., Li, Z., Liu, Z., Gong, J., Chen, X., 2024. Beyond imitation: A life-long policy learning framework for path tracking control of autonomous driving. *IEEE Trans Veh Technol*, **73**, 9786–9799.

Gou, J., Yu, B., Maybank, S. J., Tao, D., 2021. Knowledge distillation: A survey. *Int J Comput Vis*, **129**, 1789–1819.

Gunasekara, N., Pfahringer, B., Gomes, H. M., Bifet, A., 2023. Survey on online streaming continual learning. In: International Joint Conference on Artificial Intelligence, 6628–6637.

Huang, Y., Du, J., Yang, Z., Zhou, Z., Zhang, L., Chen, H., 2022. A survey on trajectory-prediction methods for autonomous driving. *IEEE Trans Intell Veh*, **7**, 652–674.

Jacobson, M. J., Wright, C. Q., Jiang, N., Rodriguez-Rivera, G., Xue, Y., 2022. Task detection in continual learning via familiarity autoencoders. In: 2022 IEEE International Conference on Systems, Man, and Cybernetics (SMC), 1–8.

Javed, K., White, M., 2019. Meta-learning representations for continual learning. In: 2019 Advances in neural information processing systems(NeurIPS), 32.

Jin, G., Liang, Y., Fang, Y., Shao, Z., Huang, J., Zhang, J., et al., 2024. Spatio-temporal graph neural networks for predictive learning in urban computing: A survey. *IEEE Trans Knowl Data Eng*, **36**, 5388–5408.

Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., Rusu, A. A., et al., 2017. Overcoming catastrophic forgetting in neural networks. *Proc Natl Acad Sci*, **114**, 3521–3526.

Lang, B., Li, X., Chuah, M. C., 2024. BEV-TP: End-to-end visual perception and trajectory prediction for autonomous driving. *IEEE Trans Intell Transp Syst*, **25**, 18537–18546.

Lee, S., Ha, J., Zhang, D., Kim, G., 2020. A Neural Dirichlet Process Mixture Model for Task-free Continual Learning. https://doi.org/10.48550/arXiv.2001.00689

Li, G., Li, Z., Knoop, V. L., van Lint, H., 2024. Unravelling uncertainty in trajectory prediction using a non-parametric approach. *Transp Res Part C Emerg Technol*, **163**, 104659.

Li, Z., Gong, C., Lin, Y., Li, G., Wang, X., Lu, C., et al., 2023. Continual driver behaviour learning for connected vehicles and intelligent transportation systems: Framework, survey and challenges. *Green Energy Intell Transp*, **2**, 100103.

Li, Z., Lin, Y., Gong, C., Wang, X., Liu, Q., Gong, J., et al., 2022. An ensemble learning framework for vehicle trajectory prediction in interactive scenarios. In: 2022 IEEE Intelligent Vehicles Symposium (IV), 51–57.

Li, Z., Lu, C., Yi, Y., Gong, J., 2021. A hierarchical framework for interactive behaviour prediction of heterogeneous traffic participants based on graph neural network. *IEEE Trans Intell Transp Syst*, **23**, 9102–9114.

Lin, Y., Li, Z., Gong, C., Liu, Q., Lu, C., Gong, J., 2023a. Rethinking trajectory prediction in real-world applications: An online task-free continual learning perspective. In: 2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC), 5020–5026.

Lin, Y., Li, Z., Gong, C., Lu, C., Wang, X., Gong, J., 2023b. Continual interactive behavior learning with traffic divergence measurement: A dynamic gradient scenario memory approach. *IEEE Trans Intell Transp Syst*, **25**, 2355–2372.

Lopez-Paz, D., Ranzato, M., 2017. Gradient episodic memory for continual learning. In: 2017 Advances in neural information processing systems(NeurIPS), 30.

Lu, Y., Wang, W., Hu, X., Xu, P., Zhou, S., Cai, M., 2022. Vehicle trajectory prediction in connected environments *via* heterogeneous context-aware graph convolutional networks. *IEEE Trans Intell Transp Syst*, **24**, 8452–8464.

Lv, K., Yuan, L., Ni, X., 2024. Learning autoencoder diffusion models of pedestrian group relationships for multimodal trajectory prediction. *IEEE Trans Instrum Meas*, **73**, 2511412.

Ma, H., Sun, Y., Li, J., Tomizuka, M., Choi, C., 2021. Continual multi-agent interaction behavior prediction with conditional generative memory. *IEEE Robot Autom Lett*, **6**, 8410–8417.

Madaan, D., Yoon, J., Li, Y., Liu, Y., Hwang, S. J., 2022. Representational Continuity for Unsupervised Continual Learning. https://doi.org/10.48550/arXiv.2110.06976

McCloskey, M., Cohen, N. J., 1989. Catastrophic interference in connectionist networks: The sequential learning problem. In: Psychology of Learning and Motivation. Amsterdam: Elsevier, 109–165.

Mehta, S. V., Patil, D., Chandar, S., Strubell, E., 2023. An empirical investigation of the role of pre-training in lifelong learning. *J Mach Learn Res*, **24**, 1–50.

Mohamed, A., Qian, K., Elhoseiny, M., Claudel, C., 2020. Social-STGCNN: A social spatio-temporal graph convolutional neural network for human trajectory prediction. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 14412–14420.

Parisi, G. I., Kemker, R., Part, J. L., Kanan, C., Wermter, S., 2019. Continual lifelong learning with neural networks: A review. *Neural Netw*, **113**, 54–71.

Parmar, J., Chouhan, S., Raychoudhury, V., Rathore, S., 2023. Open-world machine learning: Applications, challenges, and opportunities. *ACM Comput Surv*, **55**, 1–37.

Ramesh, R., Chaudhari, P., 2022. Model zoo: A Growing Brain that Learns Continually. https://doi.org/10.48550/arXiv.2106.03027

Rao, D., Visin, F., Rusu, A., Pascanu, R., Teh, Y.W., Hadsell, R., 2019. Continual unsupervised representation learning. In: 2019 Advances in neural information processing systems (NeurIPS), 32.

Rebuffi, S. A., Kolesnikov, A., Sperl, G., Lampert, C. H., 2017. iCaRL: Incremental classifier and representation learning. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 5533–5542.

Rusu, A. A., Rabinowitz, N. C., Desjardins, G., Soyer, H., Kirkpatrick, J., Kavukcuoglu, K., et al., 2016. Progressive Neural Networks. https://doi.org/10.48550/arXiv.1606.04671

Serra, J., Suris, D., Miron, M., Karatzoglou, A., 2018. Overcoming catastrophic forgetting with hard attention to the task. In: International conference on machine learning, 4548–4557.

Singh, A., 2023. Trajectory-prediction with vision: A survey. In: 2023 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW), 3310–3315.

Toneva, M., Sordoni, A., des Combes, R.T., Trischler, A., Bengio, Y., Gordon, G.J., 2019. An Empirical Study of Example Forgetting During deep Neural Network Learning. https://doi.org/10.48550/arXiv.1812.05159

van de Ven, G. M., Siegelmann, H. T., Tolias, A. S., 2020. Brain-inspired replay for continual learning with artificial neural networks. *Nat Commun*, **11**, 4069.

Wang, L., Zhang, X., Li, Q., Zhang, M., Su, H., Zhu, J., et al., 2023. Incorporating neuro-inspired adaptability for continual learning in artificial intelligence. *Nat Mach Intell*, **5**, 1356–1368.

Wang, L., Zhang, X., Su, H., Zhu, J., 2024. A comprehensive survey of continual learning: Theory, method and application. *IEEE Trans Pattern Anal Mach Intell*, **46**, 5362–5383.

Wang, L., Zhang, X., Yang, K., Yu, L., Li, C., Lanqing, H., et al., 2021. Memory Replay with Data Compression for Continual Learning. https://doi.org/10.48550/arXiv.2202.06592

Wang, W., Wang, L., Zhang, C., Liu, C., Sun, L., 2022. Social interactions for autonomous driving: A review and perspectives. *Found Trends Robot*, **10**, 198–377.

Wang, Z., Yang, E., Shen, L., Huang, H., 2025. A comprehensive survey of forgetting in deep learning beyond continual learning. *IEEE Trans Pattern Anal Mach Intell*, **47**, 1464–1483.

Wu, T., Luo, L., Li, Y. F., Pan, S., Vu, T. T., Haffari, G., 2024. Continual Learning for Large Language Models: A survey. https://arxiv.org/abs/2402.01364

Ye, F., Bors, A. G., 2023. Self-evolved dynamic expansion model for task-free continual learning. In: 2023 IEEE/CVF International Conference on Computer Vision (ICCV), 22045–22055.

Yoon, J., Yang, E., Lee, J., Hwang, S.J., 2018. Lifelong Learning with Dynamically Expandable Networks. https://doi.org/10.48550/arXiv.1708.01547

Zenke, F., Poole, B., Ganguli, S., 2017. Continual learning through synaptic intelligence. In: Proceedings of the 34th International Conference on Machine Learning, 3987–3995.

Zhan, W., Sun, L., Wang, D., Shi, H., Clausse, A., Naumann, M., et al., 2019. INTERACTION dataset: An INTERnational, Adversarial and Cooperative MoTION Dataset in Interactive Driving Scenarios with Semantic Maps. https://arxiv.org/abs/1910.03088

Zhang, E., Zhang, R., Masoud, N., 2023. Predictive trajectory planning for autonomous vehicles at intersections using reinforcement learning. *Transp Res Part C Emerg Technol*, **149**, 104063.

Zhang, S., Zhang, J., Yang, L., Chen, F., Li, S., Gao, Z., 2024. Physics guided deep learning-based model for short-term origin–destination demand prediction in urban rail transit systems under pandemic. *Engineering*, **41**, 276–296.

Huiqian Li received the B.E. degree from the School of Mechanical Engineering, Beijing Institute of Technology, China. Currently, he is a Ph.D. student at the School of Vehicle and Mobility at Tsinghua University, China. His research fields include world models and continual learning for autonomous driving.

Xiaozhou Wu received the B.S. degree in mechanics and the B.E. degree in vehicle engineering from Tsinghua University, China. Currently, he is a Ph.D. student in the School of Vehicle and Mobility at Tsinghua University. His research fields include reinforcement learning and continual learning in autonomous driving.

Jin Huang received the B.E. and Ph.D. degrees from the College of Mechanical and Vehicle Engineering, Hunan University, China. He is currently an Associate Research Professor with Tsinghua University, Beijing, China. His research interests include artificial intelligence in intelligent transportation systems, dynamics control, and fuzzy engineering.

Zhihua Zhong received the Ph.D. degree in engineering from Linkoping University, Linkoping, Sweden, in 1988. He is currently a Professor with the School of Vehicle and Mobility, Tsinghua University, China. His research interests include auto collision security technology, the punching and shaping technologies of the auto body, modularity and light-weighing auto technologies, and vehicle dynamics.