



Adaptive trajectory prediction without catastrophic forgetting

ChunYu Zhi¹ · HuaiJiang Sun¹ · Tian Xu¹

Accepted: 29 March 2023 / Published online: 19 April 2023

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2023

Abstract

Pedestrian trajectory prediction is a necessary component of autonomous driving technology. However, current methods face two troubles when utilized to the actual world, one is the distribution difference between training and testing environments, and the other is catastrophic forgetting. These two issues will lead to an inevitable drop in the overall performance of the model in real-world scenarios. To tackle these two issues, we propose a framework that consists of modules for domain adaptation and continual learning. Specifically, a pedestrian interplay modeling method based totally on pedestrian social habits is proposed. Moreover, we add a domain adaptation module to analyze the data distribution difference between the source domain and the target domain, so as to alleviate the domain difference problem. Finally, a continual learning module is introduced to retain the information which is learned to limit the change of model parameters to deal with the catastrophic forgetting. We design trajectory prediction experiments that conform to real-world activities, and the experimental results verify the superiority of our proposed model. To the best of our knowledge, we are the first work that attempts to apply domain adaptation and continual learning methods to remedy real-world trajectory prediction problems.

Keywords Trajectory prediction · Domain adaptation · Continual learning · Neural network

HuaiJiang Sun and Tian Xu contributed equally to this work.

✉ ChunYu Zhi
zhicy@njust.edu.cn

HuaiJiang Sun
sunhuaijiang@njust.edu.cn

Tian Xu
xutian@njust.edu.cn

¹ School of Computer Science and Engineering, Nanjing University of Science and Technology, XiaoLinWei, Nanjing 20094, JiangSu, China

1 Introduction

Trajectory prediction technology is becoming increasingly important with the development of intelligent society, especially in the applications of autonomous driving and surveillance systems. In autonomous driving, accurate prediction of the future trajectories of pedestrians around the vehicle can enable the vehicle to take corresponding actions in advance to avoid collisions or perform emergency braking [1, 2].

Due to the complex interactions between pedestrians and the surrounding environment, it is difficult to predict the future trajectories of pedestrians. Various factors can affect a pedestrian's trajectory, such as obstacles, other pedestrians, vehicles, traffic signals, and even the pedestrian's own subjective agency. According to [3], 70% of pedestrians tend to walk in groups. Interaction between pedestrians is primarily driven by subjective agency and social norms. The difficulty of pedestrian trajectory prediction is greatly increased due to the many factors that can influence it, such as: 1. Different social behaviors of pedestrians themselves. For example, when walking parallel to others, pedestrians maintain a group status and avoid crossing through the group when someone is walking toward them. 2. Randomness from the movement itself, as pedestrians can turn, stop, move, etc., at any time, making trajectory prediction more difficult. 3. Pedestrians can interact with surrounding objects or other pedestrians, but this interaction is too complex and subtle to accurately quantify. Due to the above reasons, the challenge of trajectory prediction is significantly increased.

To solve the above difficulties, Social-LSTM [4] designs a pooling layer to transfer the interaction information between pedestrians, and then applies a long short-term memory (LSTM) network to predict future trajectories. Following such a pattern, methods have been proposed [5–9] to share interaction information through different mechanisms, such as attention mechanisms and similarity measures. Meanwhile, in order to reflect the diversity of trajectories, some generative adversarial network (GAN) methods [10–15] learn to generate multiple feasible trajectories rather than predict a certain one. These works [16, 17] have also given good inspiration.

The above methods have some limitations, as they usually follow the principle of homogeneity between training and testing sets, i.e., training and testing are conducted on datasets with the same data distribution. Therefore, the testing results obtained in this way do not have generality and cannot adapt to the prediction of pedestrian trajectories in the real world. [18] Quantitatively and objectively evaluated the potential domain differences between the ETH and UCY experimental datasets. Table 1 provides specific numerical statistics for five trajectory domains, including the number of pedestrians, walking speed, acceleration, etc. According to this table, it is clear that there are huge differences in the number of pedestrians among the trajectory domains. In terms of pedestrian movement patterns, the average pedestrian movement speed in ETH is the highest, almost three times that of HOTEL. In addition, the average pedestrian movement acceleration in ETH is also the highest, nearly five times that of ZARA2.

Table 1 Statistics of five different scenes, ETH, HOTEL, UNIV, ZARA1, and ZARA2

Metric	Trajectory domains					E-D	S-D
	ETH	HOTEL	UNIV	ZARA1	ZARA2		
NoS	70	301	947	602	921	877	383.63
NoP	181	1053	24334	2253	5833	24153	10073.07
AN	2.586	3.498	25.696	3.743	6.333	23.11	9.78
AV (m/s)	0.437	0.178	0.205	0.369	0.206	0.259	0.11
AA (m/s ²)	0.131	0.06	0.035	0.039	0.026	0.105	0.04

NoS denotes the number of sequences to be predicted, NoP denotes the number of pedestrians, AN denotes the average number of pedestrians in each sequence, AV denotes the average velocity of pedestrians in each sequence, and AA denotes the average acceleration of pedestrians in each sequence. E-D represents extreme deviation and S-D represents standard deviation

The E-D value and S-D value also reveal huge differences between the five different trajectory domains. The situations faced in real life are undoubtedly more complex and diverse. Pedestrian trajectory prediction will face different scenarios, such as in shopping malls, where pedestrians are crowded and dense, and their trajectories are more curved, constantly changing to avoid collisions. However, on sidewalks, pedestrians are sparse and almost always walk in the same direction, and their trajectories are mostly straight lines. One case represents two different scenarios, and a network learned in a shopping mall environment may contain more redundant information for sidewalks, thus affecting the accuracy of pedestrian trajectory prediction.

To overcome the above two limitations, we add a domain adaptation module and a continual learning module to the original model. First, we substitute the aggregation layer by means of modeling pedestrian trajectories in the source and target domains as spatio-temporal graphs. Then, we use MMD (maximum mean discrepancy) to quantitatively achieve the distance between the distributions of the two datasets in the source domain and the target domain and measure the similarity of the two datasets. MMDLOSS is used to reduce training problems caused by different data distributions. Finally, a reward and punishment mechanism is delivered to the parameter training of the model through EWC regularization, so as to avoid forgetting the knowledge of the old dataset when training with the new one. Therefore, our model has both desirable generalization ability and is capable to overcome catastrophic forgetting.

In summary, our proposed model makes the following contributions in total:

1. We propose a more detailed division for pedestrian interaction, which comprehensively considers elements such as distance and direction. We believe that this division can be extensively used for pedestrian prediction.
2. We embed a domain adaptation module to enhance the learning ability of the model on new datasets by calculating the similarity of the data distribution in the target and source domains.

3. We introduce a continual learning module to preserve its potential to remember previous data via limiting parameter changes, effectively relieving the catastrophic forgetting problem of the model after studying a new dataset.
4. To the best of our knowledge, this is the first paper to propose a combination of domain adaptation and continual learning methods for trajectory prediction, in the seek of the real-world utility with deep learning-based trajectory prediction models.

2 Related work

2.1 Forecasting pedestrian trajectory

Pedestrian trajectory prediction is used to predict the future position of the target agent based on their past positions and the surrounding environment. Early research attempted to use traditional mathematical models [19] for prediction, such as Gaussian models [20, 21] and Markov decision models [22]. However, traditional mathematical models rely heavily on manually annotated prior knowledge, which is costly and lacks accuracy. With the development of deep learning, a large number of deep learning methods have been applied to solve this problem. In Social-LSTM [4], pedestrians are modeled using recurrent neural networks (RNN), and a designed pooling layer is used to integrate the hidden states of pedestrians, including shared human–machine interaction features. However, RNN-based models suffer from the problems of gradient disappearance and difficulty in scalability during the training process, so many works [3, 5–9, 23] have combined other networks to improve prediction performance. In addition, considering the subjective initiative and uncertainty of pedestrians' walking, pedestrian trajectory prediction methods based on GAN have been proposed [10–15], which introduce the idea of adversarial thinking into the task and overcome the shortcomings of previous methods that are mostly based on optimizing the distance between pedestrians and predicting only one average trajectory. Furthermore, trajectory prediction methods based on spatio-temporal graphs [24] have also been widely applied. In the prediction task, the spatio-temporal graph is divided into two dimensions: time and space, which respectively model the pedestrian's historical trajectory and simulate social interaction between pedestrians. Attention mechanisms have also been introduced into this task [25–27], and they encode the different importance of adjacent pedestrians for trajectory prediction to improve prediction accuracy. The attention mechanism breaks the sequential dependence of the RNN network and provides a more intuitive method for simulating the topological structure of pedestrians in a shared space.

2.2 Domain adaptation

Domain adaptation is a type of transfer learning that deals with the problem of different data distributions between the source and target domains. In deep learning, researchers usually assume that the training dataset and the target test dataset

have the same data distribution. However, in real life, this assumption is often difficult to satisfy. When there is a large difference in data distribution between the training dataset and the target test dataset, overfitting can easily occur, causing the trained model to perform poorly on the test dataset. To solve this problem, researchers have proposed three methods: feature adaptation, instance adaptation, and model adaptation.

Feature adaptation involves extracting the features of the source and target domains into a common feature space where the distance between the source and target domains is close enough to align them, thus improving the performance of the target domain. Instance adaptation assigns weights to the source domain data that are similar to the target domain data and uses these data to train the model, which performs relatively well on the target domain. Model adaptation finds some parameters for transfer learning to improve the performance of the target domain.

In our task, we use the first method, feature adaptation. We use a mathematical formula to measure the distance between the source and target domains and use this distance as a loss function in the deep learning network to minimize the distance and align the features of the source and target domains. Popular distance metrics include MMD [28], CORAL [29], and adversarial [30].

2.3 Continual learning

Deep learning-based AI models have achieved suitable performance, even surpassing humans on individual tasks. But deep learning models are mostly trained on static identically distributed datasets and cannot adapt or scale their behavior over time. In order to allow deep models to have the equal human-like ability to learn multiple tasks and cross-apply multiple types of knowledge, the concept of continual learning [31–33] is proposed. Ring (1997) defines continual learning as a process of continual development based on complex environments and behaviors, and the establishment of more complicated competencies on pinnacle of the that is already learned.

Continuous learning on deep neural networks has two goals: one is to deal with the catastrophic forgetting problem [34] naturally existing in neural networks due to their own design; the other is to make the training model more general, meaning that the model has the ability to learn new knowledge and memorize old knowledge at the same time. Continuous learning can be subdivided into the following four categories: (1) task-incremental CL, (2) class-incremental CL, (3) domain-incremental CL, and (4) task-agnostic continuous learning (task-agnostic CL), which is the most challenging continual learning scenario. In our work, we deal with domain incremental continual learning. It means that the data arriving at different times belong to the same category of the same task, but the data arrive in batches, and the distribution of the input data has changed. Therefore, its basic assumptions are: (1). $P(x^t) \neq P(x^{t+1})$, (2). $y^t = y^{t+1}$, (3). $P(y^t) \neq P(y^{t+1})$, with $P(\cdot)$ representing the possibility of classification.

Domain incremental continual learning is different from domain adaptation, which aims to transfer knowledge from old tasks to new tasks and only considers

the generalization ability on new tasks, while domain incremental continual learning needs to overcome catastrophic forgetting and maintain performance on old tasks as well as new ones. Our method combines the characteristics of continual learning and domain adaptation, hoping to inherit the advantages of both.

3 Problem description

Given one person i observed trajectory $V^i = \{v_1^i, \dots, v_{\text{obs}}^i\}$ from step T_1 to T_{obs} , aim to predict the future trajectory $V^i = \{v_{\text{obs}+1}^i, \dots, v_{\text{pred}}^i\}$ from step $T_{\text{obs}+1}$ to T_{pred} , where $v_t^i = (x_t^i, y_t^i) \in \mathbb{R}^2$ denote the coordinates at time t . Considering all the pedestrians in the scene, the goal is to predict trajectories of all the pedestrians simultaneously by a model $f(\cdot)$ with parameter W^* . So, the entire representation is:

$$\bar{V} = \{f(V^1, \dots, V^N | W^*)\}.. \quad (1)$$

where \bar{V} is the set of future trajectories of all the pedestrians, N evinces the number of pedestrians, and W^* represents the set of all learnable parameters in the model.

4 Our methods

4.1 Social behavior classification

In the model, we follow the Social-STGCNN model: spatio-temporal convolutional neural network (ST-GCN) and temporal extrapolator convolutional neural network (TXP-CNN). In this model, a set of spatial graphs G_t is first constructed, which represent the relative positions of pedestrians at each time step t in the scene. G_t is defined as $G_t = (V_t, E_t)$, where $V_t = v_i = (x_t^i, y_t^i), \forall i \in 1, \dots, N$ is a set of vertices of the graph G . (x_t^i, y_t^i) represent the position of the pedestrian at time $t | t \in 1, \dots, t_{\text{obs}}$. $E_t = e^{ij} | \forall i, j \in 1, \dots, n$ is a set of edges of a graph G , which represent the interaction between node i and node j . In order to model the strength of mutual influence between two nodes, an adjacency matrix A_t representing the weight relationship needs to be established.

According to the actual situation in real life, the essence of the interaction strength between pedestrians is whether the trajectories of pedestrians will intersect. Generally speaking, pedestrians will change their walking habits in two cases. One is subjective initiative, that is, the target address changes; the other is to avoid other pedestrians who may collide. For the sake of define A_t more specifically, we propose a new definition method: social behavior classification (SBC). We divide the possibilities of pedestrian trajectories into the following categories: 1. Pedestrians walk in groups. 2. Pedestrians are too far away. 3. Pedestrians walk on their backs. 4. Pedestrians walk in opposite directions without collision. 5. Pedestrians walk on opposite sides and collide. 6. Pedestrians walk in different directions without collision. 7. Pedestrians will collide when walking in different directions. In Fig. 1, we

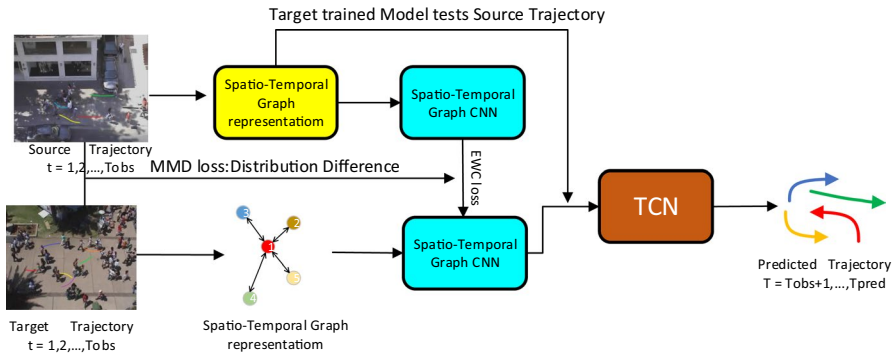


Fig. 1 Our model flowchart shows that given the trajectories of source domain and target domain, we first construct the spatial feature graph G of both through social behavior classification method, then extract the spatio-temporal node embedding of graph G through spatio-temporal graph convolution ST-GCNN, and finally predict the final trajectory through TXP-CNN. Note that when training the target domain, it is necessary to calculate the distribution difference between the source domain and the target domain first, and the magnitude of the parameter change during training should be calculated to construct an importance matrix and combine MMD loss and EWC loss to train the target domain

define that when the distance between pedestrians is less than r , it can be regarded as the group has no influence on each other. Then, we define a pedestrian circle with a radius of R , and consider the outside of the circle as infinity, and all pedestrians outside the circle have no interaction with the target agent.

For the latter cases, we introduce the concept of direction angle: α, β . When both α and β are obtuse angles, the mutual influence is 0. When both α and β are acute angles, the mutual influence is positive. But even if both α and β are acute angles, there are cases where the mutual influence is 0, such as example 4 and 6 in Fig. 2. In order to distinguish these two cases, we define a new variable γ . If the extension line in the final pedestrian speed direction has an intersection, it means that there will be a collision, $\gamma = 1$, otherwise $\gamma = 0$. So, A_t is defined as:

$$A_t = \gamma * l(v_i, v_j) * D(\alpha, \beta). \tag{2}$$

More specifically:

$$l(v_i, v_j) = \begin{cases} 0, & v_i - v_j^2 < r \text{ or } v_i - v_j^2 > R \\ \frac{1}{v_i - v_j^2}, & \text{Otherwise} \end{cases} \tag{3}$$

$$D(\alpha, \beta) = \begin{cases} \cos\alpha\cos\beta, & \frac{\pi}{2} \leq \alpha, \beta \leq \pi \\ 0, & \text{Otherwise} \end{cases} \tag{4}$$

Fig. 2 The 7 categories defined by the SBC method, v represents the walking direction of the pedestrian V_i at time T , α and β represent the angle between the walking direction and the line between the agents, and d represents the distance between the agents

4.2 Domain adaptation module

Most of the existing trajectory prediction methods are using targeted datasets for training, validation and testing. This means that the data distribution of the dataset used for training and testing is the same, which is very different from the real-life situation. To address the problem of different data distributions, we introduce a domain adaptation module into the model. We use the maximum mean discrepancy (MMD) loss function to measure the distribution distance between the source and target domains. MMD loss is defined as follows:

$$\text{MMD}(X, Y) = \frac{1}{n} \sum_{i=1}^n \phi(x_i) - \frac{1}{m} \sum_{j=1}^m \phi(y_j)^2. \quad (5)$$

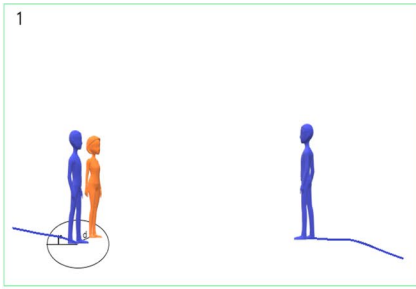
The key of MMD is how to find a suitable $\phi()$ as a mapping function, but this mapping function may be different in different tasks, and this mapping may be in a high-dimensional space, so it is difficult to select or define. In our method, we use a Gaussian kernel function: $K(u, v) = e^{-\frac{|u-v|^2}{\sigma}}$. The reason is that the Gaussian kernel can map the space of infinite latitude. For MMD loss, we have the following formula:

$$\begin{aligned} \text{MMD}(X, Y) = & \frac{1}{n^2} \sum_i^n \sum_{i'}^n (x_i, x_{i'}) - \frac{2}{nm} \sum_i^n \sum_j^m k(x_i, y_j) \\ & + \frac{1}{m^2} \sum_j^m \sum_{j'}^m k(y_j, y_{j'}) \end{aligned} \quad (6)$$

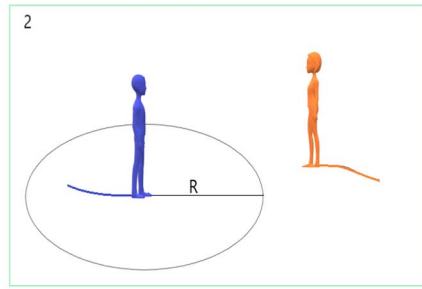
After obtaining the maximum mean difference loss, we use the loss reward and punishment mechanism to align the distribution and further strengthen the generalization ability of the model, rather than being limited to the dataset used for training, which contributes to solving the complex data distribution in real life.

4.3 Continual learning module

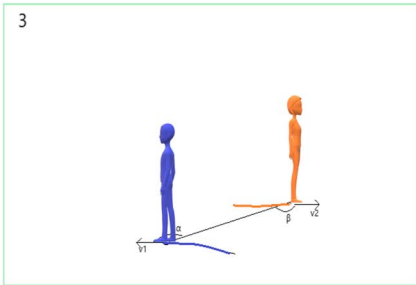
Catastrophic forgetting has always been a challenge for data-driven models. When the model is trained on a new dataset, it is difficult to guarantee that the newly trained model can still maintain the prediction accuracy of the previously trained dataset. Trajectory prediction faces complex and diverse situations in actual life. When new data enters the model processing, how to ensure knowledge retention of previously encountered situations becomes a top priority. In the training process of the neural network, the change of the parameters represents the learning ability of the network, then the parameters retain the knowledge learned on the dataset. Facing different datasets, some parameters vary greatly and some parameters vary less.



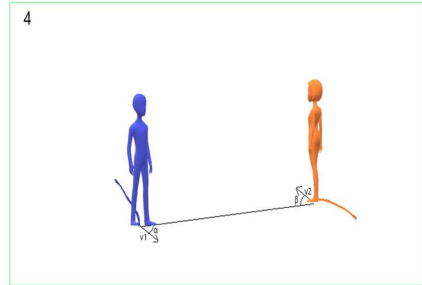
(a)



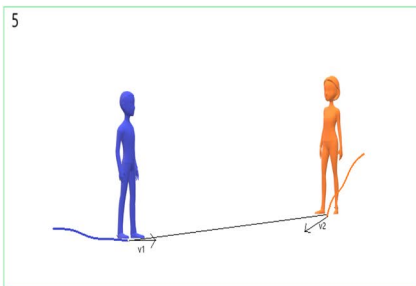
(b)



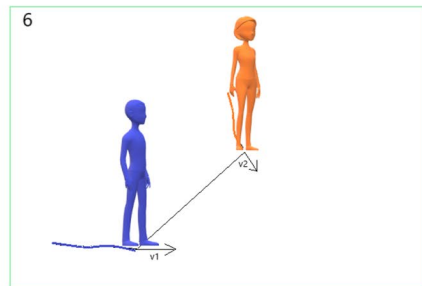
(c)



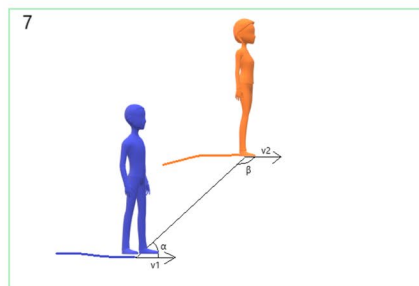
(d)



(e)



(f)



(g)

A parameter with large variation indicates that has a higher contribution to learning the dataset, while a parameter with a small change indicates that has a limited contribution to the learning of the dataset. Based on this, we believe that the previously learned knowledge can be preserved by limiting the variation of distinct parameters of the network, as shown in Fig. 3. First, we use the Fisher matrix to measure the importance of parameters. The Fisher matrix is defined as follows:

$$I(\theta) = E \left[\left(\frac{\partial}{\partial \theta} \log f(x | \theta) \right)^2 \mid \theta \right], \tag{7}$$

where θ represents learnable parameters. Actually, we can caculate the Fisher matrix by gradient. Thus, the formula can be rewritten as:

$$I(\theta) = \frac{1}{N} \sum_{(x,y) \in A} \left(\frac{\partial l_{L(\theta)(x,y)}}{\partial \theta} \right)^2, \tag{8}$$

where A represents task A , θ represents learnable parameters.

After getting the importance matrix, we can start the reward and punishment mechanism of the parameters. EWC loss can be defined as:

$$l_{\text{ewc}} = - \sum_{i=1}^{\text{params}} [I_A]_{ii} \frac{(\theta_i - [\theta_i]_A^*)^2}{2}, \tag{9}$$

where θ represents learnable parameters and I_A represents the importance matrix.

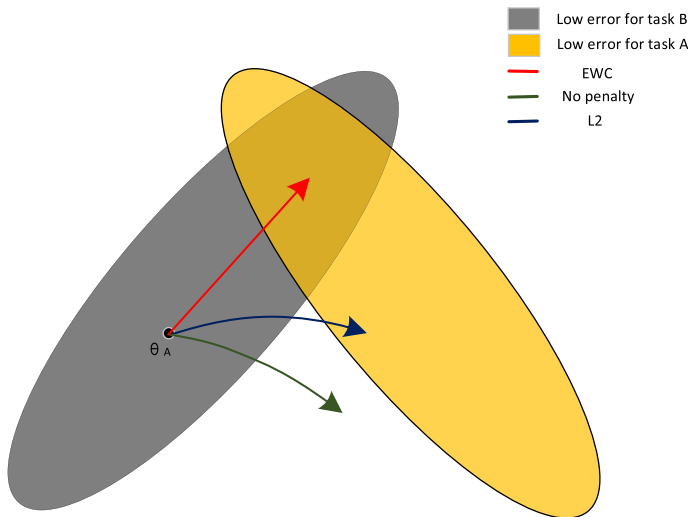


Fig. 3 On the basis of task A, the schematic diagram of using EWC regularization, L2 regularization, and penalty-free learning direction for task B, respectively. This figure appears in [35] for the first time

4.4 Objective function

The overall objective function consists of three parts, the prediction loss L_{pred} is used to predict the future trajectory prediction, the alignment loss L_{mmd} is used to align the distribution of the source trajectory domain and the target trajectory domain, and the EWC loss L_{ewc} is used to ensure the learning direction of the target domain and the source domain. The prediction loss L_{pred} is the negative log likelihood as follows:

$$L_{\text{pred}} = - \sum_{t=T_{\text{obs}}+1}^{T_{\text{pred}}} \log(P((x_t^i, y_t^i) | \hat{\mu}_t^i, \hat{\sigma}_t^i, \hat{\rho}_t^i)). \quad (10)$$

The entire model is jointly trained by $L_{\text{pred}}, L_{\text{mmd}}, L_{\text{ewc}}$, thus we have:

$$L = L_{\text{pred}} + \lambda L_{\text{mmd}} + \mu L_{\text{ewc}}, \quad (11)$$

where λ and μ are hyperparameters for balancing these three terms.

5 Experiments

In this section, we first present the definition of our proposed new setting as properly as the evaluation protocol. Then, we carry out extensive evaluations on our proposed model under this new setting, in comparison with previous existing methods.

Dataset Experiments are conducted on two real-world datasets, ETH [36] and UCY [37], as these two public datasets are widely used in this task. ETH consists of two scenes named ETH and HOTEL, and UCY consists of three scenes named UNIV, ZARA1, and ZARA2. The dataset contains a large number of interactions between pedestrians and their surroundings, including pedestrian–pedestrian interactions and pedestrian–environment interactions, such as pedestrian crossing, group and individual movements, crowd gathering and dispersal, and collision avoidance. In the scenes of the ETH dataset, most trajectories are simple straight lines, and there is not much social or spatial interaction between pedestrians. In contrast, the scenes of the UCY dataset tend to show more social interactions between pedestrians and interactions between pedestrians and the surrounding environment.

Experimental settings We introduce a new experimental setting that treats each scene as a trajectory domain. The model is first trained on one domain, then separately trained on the validation set of the other four domains, and then tested on the target and source domains, respectively. Given 5 trajectory domains, we have a total of 20 trajectory prediction tasks: $A- > B'/C'/D'/E'$, $B- > A'/C'/D'/E'$, $C- > A'/B'/C'/D'/E'$, $D- > A'/B'/C'/E'$, $E- > A'/B'/C'/D'$. Among them, A, B, C, D, and E represent ETH, HOTEL, UNIV, ZARA1, and ZARA2, respectively. This setup is somewhat challenging due to the catastrophic forgetting problem and domain differences.

Evaluation protocol To ensure a fair comparison under the new setting, an existing baseline is trained using a validation set of the source and target trajectory domains. Specifically, taking $A- > B'$ as an example, an existing baseline is trained

training set of A and validation set of B, and then evaluated on test set of A and B. Note that the validation and test sets are independent of each other, and there are no overlapping samples between them.

Baselines Five state-of-the-art methods are compared with our proposed method under the new setting and the evaluation protocol: Social-STGCNN [38], Star [24], LB-EBM [39], SGCN [40], and SocialVAE [41]. Each model corresponds to 20 tasks, for a total of 80 comparison tasks.

Evaluation metric Following two metrics are used for performance evaluation. In these two metrics, N^t is the total number of pedestrians in target trajectory domain, \bar{v}_t^i are predictions, and v_t^i are ground-truth coordinates.

- Average displacement error (ADE):

$$ADE = \frac{\sum_{i=1}^{N^t} \sum_{t=T_{\text{obs}+1}}^{T_{\text{pred}}} \|v_t^i - \bar{v}_t^i\|_2}{N^t(T_{\text{pred}} - T_{\text{obs}})}. \quad (12)$$

- Final displacement error (FDE):

$$FDE = \frac{\sum_{i=1}^{N^t} \|v_{\text{pred}}^i - \bar{v}_{\text{pred}}^i\|_2}{N^t}. \quad (13)$$

Implementation detail Similar with previous baselines, 8 frames are observed and the next 12 frames are predicted. The number of ST-GCN layers is set as 1, and TXP-CNN layers is set as 5. In the training phase, the batch size is set as 128, λ is set as 1, and μ is set as 1. The whole model is trained for 250 epochs, and Adam is applied as the optimizer. We set the initial learning rate as 0.01 and change to 0.002 after 150 epochs. In the inference phase, 20 predicted trajectories are sampled and the best among 20 predictions is used for evaluation.

5.1 Quantitative analysis

Tables 2 and 3 show the evaluation results of our method and 4 baselines on 20 tasks. From these two tables, we can see that our method outperforms these baselines in some cases. In the case where the target domain only trains the validation set, the amount of data is very different from the source domain, so the test effect of the model in the target domain will be greatly limited. However, our model has added domain adaptation. This module can better judge the data distribution of the target domain and the source domain, so our method has shown better results in ADE/FDE on the target domain. For the source domain, when the model is trained on the target domain, the learning direction will inevitably be biased toward the target domain and ignore the knowledge retention of the source domain, which is the problem of catastrophic forgetting. Even after training on the target domain with much smaller data than the source domain, all these baseline methods degrade significantly on the source domain, basically reaching the 70% drop metric. However, because the EWC module can control the parameter changes and retain the

Table 2 ADE results of our model in comparison with existing state-of-the-art baselines on 20 tasks

Method	A2B'	A2C'	A2D'	A2E'	B2A'	B2C'	B2D'	B2E'	C2A'	C2B'	C2D'	C2E'	D2A'	D2B'	D2C'	D2E'	E2A'	E2B'	E2C'	E2D'	AVG	
Target																						
SGCN	0.35	0.39	0.33	0.27	0.71	0.40	0.35	0.28	0.72	0.37	0.37	0.27	0.74	0.33	0.42	0.23	0.75	0.38	0.41	0.31	0.42	
STAR	0.29	0.61	0.49	0.41	0.63	0.67	0.53	0.42	0.67	0.31	0.50	0.39	0.59	0.34	0.63	0.39	0.63	0.29	0.70	0.55	0.50	
LB-EBM	0.17	0.34	0.23	0.20	0.36	0.36	0.32	0.21	0.35	0.26	0.37	0.24	0.33	0.22	0.35	0.18	0.33	0.17	0.30	0.26	0.29	
Social-STGCNN	0.51	0.59	0.48	0.39	0.77	0.60	0.49	0.40	0.75	0.52	0.48	0.37	0.76	0.48	0.62	0.36	0.73	0.50	0.61	0.49	0.55	
SocialVAE	0.24	0.32	0.26	0.36	0.49	0.35	0.32	0.34	0.51	0.27	0.36	0.28	0.51	0.23	0.31	0.29	0.48	0.19	0.27	0.26	0.33	
Ours	0.34	0.39	0.32	0.28	0.65	0.38	0.31	0.27	0.75	0.32	0.34	0.29	0.67	0.31	0.46	0.24	0.68	0.33	0.44	0.33	0.41	
Source																						
SGCN	0.82	0.76	0.87	0.74	0.49	0.44	0.47	0.40	0.55	0.49	0.61	0.57	0.43	0.38	0.34	0.47	0.29	0.33	0.34	0.35	0.51	
STAR	0.73	0.67	0.81	0.75	0.42	0.38	0.40	0.49	0.87	0.74	0.66	0.81	0.67	0.62	0.73	0.70	0.42	0.37	0.43	0.50	0.64	
LB-EBM	0.47	0.51	0.53	0.46	0.35	0.37	0.44	0.38	0.55	0.45	0.56	0.49	0.41	0.37	0.42	0.35	0.35	0.34	0.31	0.39	0.43	
Social-STGCNN	0.91	0.89	0.84	0.95	0.65	0.56	0.67	0.58	0.76	0.67	0.48	0.73	0.51	0.45	0.52	0.51	0.38	0.33	0.31	0.36	0.60	
SocialVAE	0.51	0.53	0.52	0.56	0.29	0.31	0.34	0.41	0.42	0.49	0.41	0.52	0.26	0.34	0.37	0.35	0.32	0.28	0.37	0.34	0.39	
Ours	0.66	0.72	0.73	0.73	0.38	0.34	0.36	0.39	0.53	0.54	0.52	0.53	0.33	0.36	0.35	0.34	0.30	0.30	0.29	0.30	0.47	

Bold represents the result with the smallest ADE/FDE under the same conditions

“2” represents from source domain to target domain. A, B, C, D, and E denote ETH, HOTEL, UNIV, ZARA1, and ZARA2, respectively

Table 3 FDE results of our model in comparison with existing state-of-the-art baselines on 20 tasks

Method	A2B'	A2C'	A2D'	A2E'	B2A'	B2C'	B2D'	B2E'	C2A'	C2B'	C2D'	C2E'	D2A'	D2B'	D2C'	D2E'	E2A'	E2B'	E2C'	E2D'	AVG	
Target																						
SGCN	0.59	0.68	0.64	0.53	1.13	0.71	0.67	0.54	1.17	0.61	0.67	0.55	1.15	0.63	0.64	0.49	1.07	0.58	0.73	0.61	0.72	
STAR	0.53	1.21	0.99	0.84	1.24	1.18	0.93	0.77	1.15	0.53	0.95	0.73	1.17	0.51	1.16	0.72	1.14	0.53	1.19	0.91	0.92	
LB-EBM	0.48	0.62	0.45	0.41	0.87	0.78	0.44	0.36	0.65	0.37	0.65	0.46	0.75	0.39	0.59	0.42	0.64	0.27	0.61	0.46	0.54	
Social-STGCNN	0.71	0.80	0.91	0.65	1.23	0.78	0.93	0.64	1.51	0.68	0.88	0.59	1.37	0.59	0.83	0.62	1.18	0.63	0.85	0.90	0.86	
SocialVAE	0.41	0.57	0.43	0.57	0.92	0.64	0.47	0.53	0.84	0.31	0.59	0.62	0.86	0.40	0.53	0.51	0.96	0.24	0.46	0.48	0.57	
Ours	0.57	0.69	0.51	0.47	1.16	0.65	0.53	0.43	1.48	0.41	0.56	0.49	1.30	0.44	0.67	0.41	1.08	0.53	0.68	0.56	0.68	
Source																						
SGCN	1.54	1.61	1.43	1.48	0.64	0.57	0.72	0.55	0.93	0.87	0.98	0.84	0.94	0.87	0.92	0.79	0.53	0.59	0.52	0.49	0.89	
STAR	1.63	1.75	1.52	1.57	0.62	0.61	0.66	0.59	1.51	1.43	1.56	1.50	1.26	1.13	1.08	1.28	0.79	0.84	0.88	0.76	1.15	
LB-EBM	0.87	0.96	0.90	0.85	0.52	0.67	0.60	0.54	0.85	0.88	0.96	0.94	0.58	0.64	0.57	0.62	0.49	0.57	0.55	0.54	0.71	
Social-STGCNN	1.61	1.72	1.68	1.57	0.71	0.65	0.69	0.59	1.13	1.19	0.97	1.10	0.90	0.79	0.88	0.88	0.58	0.56	0.55	0.58	0.97	
SocialVAE	0.92	0.98	1.03	0.97	0.49	0.56	0.59	0.47	0.62	0.59	0.67	0.64	0.57	0.46	0.49	0.51	0.54	0.52	0.62	0.59	0.64	
Ours	1.25	1.35	1.42	1.30	0.56	0.53	0.56	0.53	0.97	0.97	0.91	0.93	0.54	0.60	0.57	0.56	0.52	0.51	0.52	0.50	0.78	

Bold represents the result with the smallest ADE/FDE under the same conditions

“2” represents from source domain to target domain. A, B, C, D, and E denote ETH, HOTEL, UNIV, ZARA1, and ZARA2, respectively

Table 4 ADE/FDE results obtained by replacing the kernel function with the SBC method in Social-STGCNN

Method	eth	Hotel	Univ	Zara1	Zara2	AGV
Social-STGCNN	0.64 /1.11	0.49/0.85	0.44/0.79	0.34/ 0.53	0.30/0.48	0.44/0.75
Social-STGCNN/SBC	0.65/ 1.10	0.33/0.45	0.39/0.68	0.30 /0.56	0.27/0.44	0.39/0.65

Bold represents the result with the smallest ADE/FDE under the same conditions

knowledge of the source domain, our model still maintains good insurance in the source domain, and the reduction index is controlled within 15%.

To verify the effectiveness of our proposed SBC method, we replace the kernel function in Social-STGCNN with SBC, and the results are shown in Table 4. The results show that our SBC method has a positive effect on pedestrian trajectory prediction. In most cases, we outperform the original kernel function, which means that distance is not the only factor that affects pedestrian interaction, and pedestrians' social habits, etc., also affect pedestrian trajectories.

To verify the effectiveness of each module, we conducted further ablation experiments on the network structure of this paper. We used the ETH dataset as the source dataset and the hotel dataset as the target dataset. After training on the source dataset, we obtained the ADE/FDE results of the test set. Then, we trained on the validation set of the target dataset and tested on the test set to verify the effectiveness of the domain adaptation module. Finally, we tested the performance of the model on the test set of the source dataset to verify the effectiveness of the continual learning module.

As shown in Table 5, compared with 1.2, the network structure with the domain adaptation module achieved better performance on the target dataset. However, due to the existence of the domain adaptation module, the network parameters quickly approached the target dataset, resulting in severe knowledge loss on the source dataset and leading to a significant drop in performance when returning to train on the source dataset. Compared with 1.3, the continual learning module effectively mitigated the catastrophic forgetting problem. After training on the target dataset, the network still maintained good predictive performance on the source dataset. Compared with 1.2.3.4, we found that the network could achieve

Table 5 ADE/FDE results of domain adaptation module and continuous learning module ablation experiments

Number	Model components		Source dataset 1	Target dataset	Source dataset 2
	Domain adaptation	Continual learning			
1	No	No	0.70/1.19	0.58/0.81	1.32/2.44
2	Yes	No	0.71/1.31	0.33/0.45	2.01/3.74
3	No	Yes	0.73/1.24	0.38/0.50	0.76/1.31
4	Yes	Yes	0.70/1.21	0.35/0.42	0.75/1.28

our expected results, that is, to maintain the predictive performance on the source dataset while improving the predictive performance on the target dataset with the combined action of the domain adaptation and continual learning module.

6 Conclusion

In this paper, we propose a unified model that incorporates graph neural networks and temporal convolutional neural networks for future trajectory prediction and adds domain adaptation and continual learning modules to mitigate domain differences and catastrophic forgetting. Extensive experiments demonstrate the superiority of our model in future trajectory prediction. Our work is the first to combine domain adaptation and continual learning to study future trajectory prediction, making an appropriate attempt for the application of trajectory prediction based on deep learning in real life. In the future work, we will conduct more in-depth research on the factors affecting pedestrian trajectory.

Acknowledgements This work was supported in part by the National Natural Science Foundation of China (NO.62176125, 61772272).

Author contributions All authors have made contributions to the final draft of this article.

Data availability The datasets generated during and/or analyzed during the current study are available from the corresponding author on reasonable request.

Declarations

Conflict of interest All work was completed in Nanjing University of Science and Technology, and there was no other object of interest competition.

References

1. Luo Y, Cai P, Bera A, Hsu D, Lee WS, Manocha D (2018) Porca: Modeling and planning for autonomous driving among many pedestrians. *IEEE Robot Autom Lett* 3(4):3418–3425
2. Raksincharoensak P, Hasegawa T, Nagai M (2016) Motion planning and control of autonomous driving intelligence system based on risk potential optimization framework. *Int J Automot Eng* 7(AVEC14):53–60
3. Bisagno N, Zhang B, Conci N (2018) Group lstm: group trajectory prediction in crowded scenarios. In: *Proceedings of the European Conference on Computer Vision (ECCV) workshops*, pp 0–0
4. Alahi A, Goel K, Ramanathan V, Robicquet A, Fei–Fei L, Savarese S (2016) Social lstm: human trajectory prediction in crowded spaces. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp 961–971
5. Hu Y, Chen S, Zhang Y, Gu X (2020) Collaborative motion prediction via neural motion message passing. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp 6319–6328
6. Zhu Y, Ren D, Xu Y, Qian D, Fan M, Li X, Xia H (2021) Simultaneous past and current social interaction-aware trajectory prediction for multiple intelligent agents in dynamic scenes. *ACM Trans Intell Syst Technol (TIST)* 13(1):1–16

7. Liang J, Jiang L, Niebles JC, Hauptmann AG, Fei-Fei L (2019) Peeking into the future: predicting future person activities and locations in videos. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 5725–5734
8. Xu Y, Yang J, Du S (2020) Cf-lstm: cascaded feature-based long short-term networks for predicting pedestrian trajectory. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol 34, pp 12541–12548
9. Zhang P, Ouyang W, Zhang P, Xue J, Zheng N (2019) Sr-lstm: state refinement for lstm towards pedestrian trajectory prediction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 12085–12094
10. Chen G, Li J, Lu J, Zhou J (2021) Human trajectory prediction via counterfactual analysis. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 9824–9833
11. Chen G, Li J, Zhou N, Ren L, Lu J (2021) Personalized trajectory prediction via distribution discrimination. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 15580–15589
12. Cheng H, Liao W, Tang X, Yang MY, Sester M, Rosenhahn B (2021) Exploring dynamic context for multi-path trajectory prediction. In: 2021 IEEE International Conference on Robotics and Automation (ICRA), pp 12795–12801. IEEE
13. Salzmann T, Ivanovic B, Chakravarty P, Pavone M (2020) Trajectron++: dynamically-feasible trajectory forecasting with heterogeneous data. In: European Conference on Computer Vision, pp 683–700. Springer
14. Shafiee N, Padir T, Elhamifar E (2021) Introvert: human trajectory prediction via conditional 3d attention. In: Proceedings of the IEEE/cvf Conference on Computer Vision and Pattern recognition, pp 16815–16825
15. Xu Y, Ren D, Li M, Chen Y, Fan M, Xia H (2021) Tra2tra: trajectory-to-trajectory prediction with a global spatial-temporal attentive neural network. IEEE Robot Autom Lett 6(2):1574–1581
16. Wu H, Nie J, He Z, Zhu Z, Gao M (2022) One-shot multiple object tracking in uav videos using task-specific fine-grained features. Remote Sens. <https://doi.org/10.3390/rs14163853>
17. Wu H, He Z, Gao M (2023) Gcevt: learning global context embedding for vehicle tracking in unmanned aerial vehicle videos. IEEE Geosci Remote Sens Lett 20:1–5. <https://doi.org/10.1109/LGRS.2022.3228527>
18. Xu Y, Wang L, Wang Y, Fu Y (2022) Adaptive trajectory prediction via transferable gnn. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 6520–6531
19. Ma Q, Olshevsky A (2020) Adversarial crowdsourcing through robust rank-one matrix completion. Adv Neural Inf Process Syst 33:21841–21852
20. Ellis D, Sommerlade E, Reid I (2009) Modelling pedestrian trajectory patterns with gaussian processes. In: 2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops, pp 1229–1234. IEEE
21. Tay MKC, Laugier C (2008) Modelling smooth paths using gaussian processes. In: Field and service robotics, pp 381–390. Springer
22. Kitani KM, Ziebart BD, Bagnell JA, Hebert M (2012) Activity forecasting. In: European Conference on Computer Vision, pp 201–214. Springer
23. Bisagno N, Saltori C, Zhang B, De Natale FG, Conci N (2021) Embedding group and obstacle information in lstm networks for human trajectory prediction in crowded scenes. Comput Vis Image Underst 203:103126
24. Yu C, Ma X, Ren J, Zhao H, Yi S (2020) Spatio-temporal graph transformer networks for pedestrian trajectory prediction. In: European Conference on Computer Vision, pp 507–523. Springer
25. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I (2017) Attention is all you need. Adv Neural Inf Process Syst, **30**
26. Liu Y, Zhang J, Fang L, Jiang Q, Zhou B (2021) Multimodal motion prediction with stacked transformers. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 7577–7586
27. Bertugli A, Calderara S, Coscia P, Ballan L, Cucchiara R (2021) Ac-vrnn: attentive conditional-vrnn for multi-future trajectory prediction. Comput Vis Image Underst 210:103245
28. Ni J, Qiu Q, Chellappa R (2013) Subspace interpolation via dictionary learning for unsupervised domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 692–699
29. Zhuo J, Wang S, Zhang W, Huang Q (2017) Deep unsupervised convolutional domain adaptation. In: Proceedings of the 25th ACM International Conference on Multimedia, pp 261–269

30. Wu M, Pan S, Zhou C, Chang X, Zhu X (2020) Unsupervised domain adaptive graph convolutional networks. In: Proceedings of The Web Conference 2020, pp 1457–1467
31. De Lange M, Aljundi R, Masana M, Parisot S, Jia X, Leonardis A, Slabaugh G, Tuytelaars T (2019) Continual learning: a comparative study on how to defy forgetting in classification tasks. arXiv preprint [arXiv:1909.08383](https://arxiv.org/abs/1909.08383)
32. Parisi GI, Kemker R, Part JL, Kanan C, Wermter S (2019) Continual lifelong learning with neural networks: a review. *Neural Netw* 113:54–71
33. Lesort T, Lomonaco V, Stoian A, Maltoni D, Filliat D, Díaz-Rodríguez N (2020) Continual learning for robotics: definition, framework, learning strategies, opportunities and challenges. *Inf fusion* 58:52–68
34. Michael McCloskey (1989) Catastrophic interference in connectionist networks: the sequential learning problem. *Psychol Learn Motiv* 24:109–165
35. Huszár F (2017) On quadratic penalties in elastic weight consolidation. arXiv preprint [arXiv:1712.03847](https://arxiv.org/abs/1712.03847)
36. Pellegrini S, Ess A, Schindler K, Van Gool L (2009) You'll never walk alone: modeling social behavior for multi-target tracking. In: 2009 IEEE 12th International Conference on Computer Vision, pp 261–268. IEEE
37. Lerner A, Chrysanthou Y, Lischinski D (2007) Crowds by example. In: *Computer graphics forum*, vol 26, pp 655–664. Wiley Online Library
38. Mohamed A, Qian K, Elhoseiny M, Claudel C (2020) Social-stgcn: a social spatio-temporal graph convolutional neural network for human trajectory prediction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 14424–14432
39. Pang B, Zhao T, Xie X, Wu YN (2021) Trajectory prediction with latent belief energy-based model. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 11814–11824
40. Shi L, Wang L, Long C, Zhou S, Zhou M, Niu Z, Hua G (2021) Sgcn: sparse graph convolution network for pedestrian trajectory prediction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 8994–9003
41. Xu P, Hayet J-B, Karamouzas I (2022) Socialvae: human trajectory prediction using timewise latents. *Computer Vision - ECCV 2022*. Springer, Cham, pp 511–528

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.