



An Attention-Based Interaction-Aware Spatio-Temporal Graph Neural Network for Trajectory Prediction

Hao Zhou^{1,4}, Dongchun Ren², Huaxia Xia², Mingyu Fan^{2,3}, Xu Yang⁴(✉),
and Hai Huang¹(✉)

¹ National Key Laboratory of Science and Technology of Underwater Vehicle,
Harbin Engineering University, Harbin 150001, China

zhouhao94@yahoo.com, haihus@163.com

² Meituan-Dianping Group, Beijing 100102, China

{rendongchun,xiahuaxia}@meituan.com

³ School of Computer Science and Artificial Intelligence, Wenzhou University,
Wenzhou 325035, China

fanmingyu@wzu.edu.cn

⁴ State Key Laboratory of Management and Control for Complex System,
Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China

xu.yang@ia.ac.cn

Abstract. Pedestrian trajectory prediction in crowd scenes is very useful in many applications such as video surveillance, self-driving cars, and robotic systems; however, it remains a challenging task because of the complex interactions and uncertainties of crowd motions. In this paper, a novel trajectory prediction method called the Attention-based Interaction-aware Spatio-temporal Graph Neural Network (AST-GNN) is proposed. AST-GNN uses an Attention mechanism to capture the complex interactions among multiple pedestrians. The attention mechanism allows for a dynamic and adaptive summary of the interactions of the nearby pedestrians. When the attention matrix is obtained, it is formulated into a propagation matrix for graph neural networks. Finally, a Time-extrapolator Convolutional Neural Network (TXP-CNN) is used in the temporal dimension of the aggregated features to predict the future trajectories of the pedestrians. Experimental results on benchmark pedestrian datasets (ETH and UCY) reveal the competitive performances of AST-GNN in terms of both the final displacement error (FDE) and average displacement error (ADE) as compared with state-of-the-art trajectory prediction methods.

Keywords: Trajectory prediction · Spatio-temporal prediction · Graph neural networks

1 Introduction

Pedestrian trajectory prediction in crowd scenes is important in many applications including robotic systems, video surveillance, and self-driving cars.

© Springer Nature Switzerland AG 2020

H. Yang et al. (Eds.): ICONIP 2020, CCIS 1333, pp. 38–45, 2020.

https://doi.org/10.1007/978-3-030-63823-8_5

Accurate trajectory prediction in surveillance systems is helpful for the identification of suspicious activities. When applied to robotics and self-driving cars, it enables the controller to make intelligent strategies in advance of some critical situations, such as emergency braking or collision avoidance.

Early pedestrian trajectory prediction methods, such as the Gaussian process regression method [14], the kinematic and dynamic method [16], and the Bayesian network method [7], ignore the interactions among pedestrians and are only able to make reasonable short-term predictions. As discussed in [1], pedestrian trajectory prediction is a challenging task because of the complex interactions among pedestrians, which are referred to as social behavior. Pedestrians tend to move in groups and avoid collisions when walking in the opposite directions, and their interactions are roughly driven by common sense and social conventions. Because their destinations and possible paths are unknown, the motion of multiple pedestrians in a crowd scene is generally randomly distributed. The GRIP method [9] proposes the use of a graph neural network (GNN) for trajectory prediction. However, the graph is constructed via reference to the Euclidean distance between agents and is not the optimal choice because all neighbors are treated equally.

Instead of the restriction of the local neighborhood assumption, the attention mechanism is helpful for encoding the relative influences and the potential spatial interactions among pedestrians due to the unequal importance of the neighboring pedestrians that contribute to the trajectory prediction. In this paper, the use of the graph attention (GAT) [17] mechanism is proposed to capture the interactions among pedestrians and then formulate the information into a propagation matrix for a GNN [18]. Because the GNN is able to define a normalized weighted aggregation of features, it is a powerful tool with which to combine the interactions and make a reasonable prediction. With the features aggregated by the GNN, a Time-extrapolator Convolutional Neural Network (TXP-CNN) is used as the decoder for prediction in the temporal dimension of data.

The remainder of this paper is organized as follows. A brief overview of related work is provided in Sect. 2, and the proposed prediction model is defined and presented in Sect. 3. Experimental comparisons with state-of-the-art methods on the ETH [12] and UCY [8] pedestrian datasets are presented in Sect. 4. Finally, some concluding remarks are given in Sect. 5.

2 Related Works

A recent study [1] indicates that the recurrent neural network (RNN) and its variants, namely long short-term memory (LSTM) and gated recurrent units (GRUs), are successful in trajectory prediction. Based on the multi-modal distribution assumption, Social-GAN [4] extends the social LSTM into an RNN-based generative model. The CIDNN method [19] uses motion features extracted by LSTM networks to encode the interactions among agents. Peek into the future (PIF) [10] and Sophie [15] use deep convolutional neural networks (CNNs) to extract the visual features from the scene and combines the motion features into

LSTMs for scene compliant trajectory prediction. Alternatively, [2] uses temporal convolutional networks to encode or decode the trajectories.

Many prediction methods propose the use of attention models to automatically assign importance to nodes. The social-BiGAT [6] method uses a graph attention model to capture the interactions between pedestrians and the surrounding scene. The STGAT method [5] first uses an LSTM to capture the trajectory information of each agent and applies GAT to model the interactions of multiple agents at every time step. Recently, the VectorNet method [3] has been proposed and utilizes a self-attention mechanism to aggregate all motion features of road agents. Social-STGCNN [11] defines a spatial graph by a Euclidean distance based kernel function. As compared with Social-STGCNN, the attention-based adaptive graph, rather than the distance-based graph [11], is used in the proposed method.

3 The Proposed Scheme

To overcome the weak graph representation issue of Social-STGCNN [11], the novel Attention-based Spatio-temporal GNN (AST-GNN) is proposed for pedestrian trajectory prediction in this section. The model is described in three parts, namely: (1) attention-based spatial graph representation, (2) the attention-based spatial GNN model, and (3) the time-extrapolator trajectory prediction model. The architecture of the proposed AST-GNN scheme is illustrated in Fig. 1.

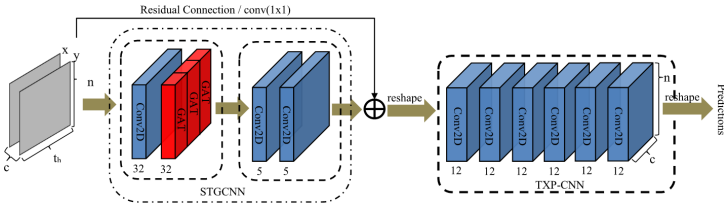


Fig. 1. The architecture of the proposed AST-GNN scheme.

3.1 Attention-Based Spatial Graph Representation

Input Representation of Pedestrian Prediction. The original trajectory data are sparse, so the raw data are first converted into a format that is suitable for subsequent efficient computation. Assuming that n pedestrians in a scene were observed in the past t time steps, this information is represented in a 3D array input with a size of $(n \times t \times c)$, where $c = 2$ denotes the coordinates (x_t^i, y_t^i) of a pedestrian.

Graph Representation of Pedestrian Prediction. The graph for pedestrian trajectory prediction is constructed in the spatial dimension. At time t , a spatial graph G_t is constructed that represents the relative locations of pedestrians in a scene at time step t . G_t is defined as $G_t = \{V_t, E_t\}$, where $V_t = \{v_t^i \mid \forall i \in \{1, \dots, N\}\}$ is a node set of pedestrians in a scene. The feature vector of v_t^i on a node is the coordinates of the i -th pedestrian at time step t . $E_t = \{e_t^{ij} \mid \forall i, j \in \{1, \dots, N\}\}$ is the edge set within graph G_t , e_t^{ij} denotes the edge between v_t^i and v_t^j .

To model how strongly two nodes influence each other, a weighted adjacency matrix is used to replace the normal adjacency matrix. In general, the distance relationship between pedestrians is used to build the weight of an adjacency matrix. However, the social network of a person is a complex problem, and cannot simply be decided by the distances between a pedestrian and the other. Thus, in this work, the GAT mechanism is used to adaptively learn the weighted adjacency matrix.

Graph Attention Mechanism. The GAT mechanism is used to calculate the weighted adjacency matrix A_t at time step t . The input of GAT mechanism $H_t = \{h_t^i \mid h_t^i \in \mathbb{R}^F, \forall i \in \{1, \dots, N\}\}$ is the set of all feature vectors of nodes at time step t . To obtain sufficient expressive power to transform the input features into higher-level features, a learnable linear transformation $\mathbf{W} \in \mathbb{R}^{F' \times F}$ is used to transform feature vectors from \mathbb{R}^F to $\mathbb{R}^{F'}$. Then, the self-attention mechanism is performed on the nodes:

$$\alpha_t^{ij} = \frac{\exp(\text{LeakyReLU}(\mathbf{a}^T [\mathbf{W}h_t^i \parallel \mathbf{W}h_t^j]))}{\sum_{k \neq i} \exp(\text{LeakyReLU}(\mathbf{a}^T [\mathbf{W}h_t^i \parallel \mathbf{W}h_t^k]))}. \quad (1)$$

where α_t^{ij} measures the impact of the j -th node on the i -th node at time step t , $\mathbf{a} \in \mathbb{R}^{2F'}$ is a weight vector, \cdot^T represents transposition, and \parallel represents the concatenation operator. It should be noted that the activation function LeakyReLU uses the negative input slope $\alpha = 0.2$.

3.2 Attention-Based Spatial GNN Model

In the proposed AST-GNN model, the GAT mechanism is added to adaptively learn the weighted adjacency matrix. As described in Fig. 1, the AST-GNN consists of two parts, namely the spatial graph convolutional block and the temporal convolutional block. Moreover, a residual connection is used to connect the input and output to avoid significant information loss.

Spatial Graph Neural Networks. As described in Sect. 3.1, the input data format is $(n \times t \times c)$, and the attribute of each node is the coordinates of pedestrians. A convolutional layer with a kernel size 1 is first used to extract convolutional feature maps f_{conv}^t . Then, the attention-based graph representation operator presented in Sect. 3.1 is used to construct the weighted adjacency matrix A_t

using feature maps f_{conv}^t . The normalized weighted adjacency matrix A_t is then used to perform the graph operation by multiplication with f_{conv}^t as follow:

$$f_{graph}^t = \sigma(\Lambda_t^{-\frac{1}{2}} \hat{A}_t \Lambda_t^{-\frac{1}{2}} f_{conv}^t). \quad (2)$$

where f_{graph}^t is the graph feature map at time step t , $\hat{A}_t = A_t + I$, Λ_t is the diagonal matrix of \hat{A}_t , and σ is the activation function of the parametric ReLU (PReLU).

Time-Extrapolator Trajectory Prediction Model. The temporal convolutional block is used to model the graph information in the time dimension. First, the outputs of spatial graph convolutional blocks at different time steps are stacked into feature V with the format $(n \times t \times c_1)$, where $c_1 = 32$ is the feature dimension. Then, a convolutional layer with a kernel size of 1 is used to reduce feature dimension from c_1 to c_2 for subsequent efficient computation, where $c_2 = 5$. A convolutional layer with a kernel size of (1×3) is then used to process the graph feature along the temporal dimension. Finally, a residual connection between the input and output is used to produce the graph embedding \tilde{V} .

3.3 Trajectory Prediction Model

As illustrated in Fig. 1, an encoder-decoder model is adopted to predict the trajectories of all pedestrians in a scene. The AST-GNN model is used as the encoder, and the Time-extrapolator Convolutional Neural Network (TXP-CNN) is the decoder. As presented in Fig. 1, the model first extracts the spatial node embedding \tilde{V} from the input graph. Then, the TXP-CNN receives \tilde{V} features and produces the predicted trajectories of pedestrians.

Time-Extrapolator Convolutional Neural Network. The TXP-CNN receives the graph embedding \tilde{V} and operates directly in the temporal dimension. The graph embedding \tilde{V} has a shape of $(n \times t \times c_2)$, we first reshape the features into the format $(n \times c_2 \times t)$. Then, five convolutional layers with kernel sizes of (3×1) are used to operator in the reshaped features, and PReLU activation function is added along every convolution operator. Next, a convolutional layer with kernel size of (3×1) is used to produce the output feature with format $(n \times c_2 \times t_f)$, where $t_f = 12$ is the expected prediction time steps. Finally, we reshape the output feature into format $(n \times c_2 \times t_f)$ and feed the reshaped feature into a GMM model for predicting future trajectories.

4 Experiments

4.1 Datasets and Metrics

In this section, the proposed method is evaluated on two well-known pedestrian trajectory prediction datasets: namely ETH [12] and UCY [8]. ETH contains

two scenes respectively denoted as ETH and HOTEL, while UCY contains three scenes respectively denoted as ZARA1, ZARA2, and UNIV. The samples in both datasets were sampled at 0.4 s over 8 s. For a fair comparison with other methods, the experimental setups of the proposed method followed that of social-LSTM [1]. During training and evaluation, the first 3.2 s (8 frames) were used as the observed history and the remaining 4.8 s (12 frames) were considered as the prediction ground truth.

Two common metrics were used for evaluation, namely the average displacement error (ADE) [13] and final displacement error (FDE) [1]. The ADE measures the average prediction performance along the trajectory, while the FDE considers only the prediction precision at the end points.

4.2 Implementation Details

The PyTorch deep learning framework was used to implement the proposed network. The models were trained with an Nvidia Tesla V100 GPU. The stochastic gradient descent (SGD) algorithm was used as the optimizer. The model was trained for 250 epochs with a batch size of 128. The initial learning rate was set to 0.01 and the decay is set to 0.002 after 150 epochs.

4.3 Comparison with the State-of-the-art Methods

As exhibited in Table 1, the proposed method was compared with other state-of-the-art methods on the ETH and UCY dataset in terms of the ADE/FDE metrics. As can be seen, the proposed AST-GNN method achieved new state-of-the-art performance and outperformed all existing state-of-the-art methods in terms of the FDE metric. This improvement is attributable to the added GAT mechanism. Regarding the FDE metric, the proposed method achieved an error of 0.74 with a 20% decrease as compared to the recent state-of-the-art method SR-LSTM-2 [21]. Regarding the ADE metric, the error of the proposed method was slightly greater than that of SR-LSTM-2 by 4%, but it was still one of the best results. More remarkably, the proposed method, which does not use scene image information, outperformed methods that utilized image information, such as SR-LSTM, PIF and Sophie.

5 Conclusion

In this paper, a novel AST-GNN method was proposed that learns representative, robust, and discriminative graph embedding for pedestrians trajectory prediction. In the proposed method, the GAT mechanism is used to adaptively learn the weighted adjacency matrix, which enhances the graph representation ability. The results of experiments on the ETH and UCY datasets demonstrate that the proposed method outperformed existing pedestrian trajectory prediction methods. In the future, the GAT mechanism will be further used on a temporal graph of a pedestrian trajectory prediction model to enhance the representation ability.

Table 1. Comparison with state-of-the-art methods in term of the ADE/FDE metrics. The best performance for each dataset is highlighted in bold. * indicates non-probabilistic models.

Method	ETH	HOTEL	UNIV	ZARA1	ZARA2	AVG
Linear * [1]	1.33/2.94	0.39/0.72	0.82/1.59	0.62/1.21	0.77/1.48	0.79/1.59
SR-LSTM-2 * [21]	0.63/1.25	0.37/0.74	0.51/1.10	0.41/0.90	0.32/0.70	0.45/0.94
S-LSTM [1]	1.09/2.35	0.79/1.76	0.67/1.40	0.47/1.00	0.56/1.17	0.72/1.54
S-GAN-P [4]	0.87/1.62	0.67/1.37	0.76/1.52	0.35/0.68	0.42/0.84	0.61/1.21
Sophie [15]	0.70/1.43	0.76/1.67	0.54/1.24	0.30/0.63	0.38/0.78	0.54/1.15
PIF [10]	0.73/1.65	0.30/0.59	0.60/1.27	0.38/0.81	0.31/0.68	0.46/1.00
STSGCN [20]	0.75/1.63	0.63/1.01	0.48/1.08	0.30/0.65	0.26/0.57	0.48/0.99
GAT [6]	0.68/1.29	0.68/1.40	0.57/1.29	0.29/0.60	0.37/0.75	0.52/1.07
Social-BiGAN [6]	0.69/1.29	0.49/1.01	0.55/1.32	0.30/0.62	0.36/0.75	0.48/1.00
Social-STGCNN [11]	0.75/1.35	0.47/0.84	0.49/0.90	0.39/0.62	0.34/0.52	0.49/0.85
The proposed	0.69/1.27	0.36/0.62	0.46/0.83	0.32/0.53	0.28/ 0.44	0.42/0.74

Acknowledgments. This work is supported partly by the National Natural Science Foundation (NSFC) of China (grants 61973301, 61972020, 61633009, 51579053, 61772373 and U1613213), partly by the National Key R&D Program of China (grants 2016YFC0300801 and 2017YFB1300202), partly by the Field Fund of the 13th Five-Year Plan for Equipment Pre-research Fund (No. 61403120301), partly by Beijing Science and Technology Plan Project, partly by the Key Basic Research Project of Shanghai Science and Technology Innovation Plan (No. 15JC1403300), partly by Beijing Science and Technology Project. (No. Z181100008918018), partly by Beijing Nova Program (No. Z201100006820046), and partly by Meituan Open R&D Fund.

References

1. Alahi, A., Goel, K., Ramanathan, V., Robicquet, A., Li, F., Savarese, S.: Social LSTM: human trajectory prediction in crowded spaces. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2016
2. Cui, H., et al.: Multimodal trajectory predictions for autonomous driving using deep convolutional networks. In: 2019 International Conference on Robotics and Automation (ICRA), pp. 2090–2096, May 2019
3. Gao, J., et al.: VectorNet: encoding HD maps and agent dynamics from vectorized representation. ArXiv abs/2005.04259 (2020)
4. Gupta, A., Johnson, J., Fei-Fei, L., Savarese, S., Alahi, A.: Social GAN: socially acceptable trajectories with generative adversarial networks. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2255–2264, June 2018
5. Huang, Y., Bi, H., Li, Z., Mao, T., Wang, Z.: STGAT: modeling spatial-temporal interactions for human trajectory prediction. In: The IEEE International Conference on Computer Vision (ICCV), October 2019
6. Kosaraju, V., Sadeghian, A., Mart m-Mart m, R., Reid, I., Rezatofighi, H., Savarese, S.: Social-BiGAT: multimodal trajectory forecasting using bicycle-GAN and graph attention networks. In: Advances in Neural Information Processing Systems, vol. 32, pp. 137–146 (2019)

7. Lefevre, S., Laugier, C., Ibanezguzman, J.: Exploiting map information for driver intention estimation at road intersections. In: 2011 IEEE Intelligent Vehicles Symposium (IV), pp. 583–588, June 2011
8. Lerner, A., Chrysanthou, Y., Lischinski, D.: Crowds by example. *Comput. Graph. Forum* **26**(3), 655–664 (2007)
9. Li, X., Ying, X., Chuah, M.C.: GRIP: graph-based interaction-aware trajectory prediction. In: 2019 IEEE Intelligent Transportation Systems Conference (ITSC), pp. 3960–3966 (2019)
10. Liang, J., Jiang, L., Niebles, J.C., Hauptmann, A.G., Fei-Fei, L.: Peeking into the future: predicting future person activities and locations in videos. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5718–5727, June 2019
11. Mohamed, A., Qian, K., Elhoseiny, M., Claudel, C.: Social-STGCNN: a social spatio-temporal graph convolutional neural network for human trajectory prediction. arXiv e-prints [arXiv:2002.11927](https://arxiv.org/abs/2002.11927) (2020)
12. Pellegrini, S., Ess, A., Schindler, K., van Gool, L.: You’ll never walk alone: modeling social behavior for multi-target tracking. In: 2009 IEEE 12th International Conference on Computer Vision, pp. 261–268, September 2009
13. Pellegrini, S., Ess, A., Van Gool, L.: Improving data association by joint modeling of pedestrian trajectories and groupings. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010. LNCS, vol. 6311, pp. 452–465. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-15549-9_33
14. Rasmussen, C.E., Williams, C.K.I.: Gaussian Processes for Machine learning (Adaptive Computation and Machine Learning). The MIT Press, Cambridge (2005)
15. Sadeghian, A., Kosaraju, V., Sadeghian, A., Hirose, N., Rezaatoughi, H., Savarese, S.: SoPhie: an attentive GAN for predicting paths compliant to social and physical constraints. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
16. Toledo-Moreo, R., Zamora-Izquierdo, M.A.: Imm-based lane-change prediction in highways with low-cost gps/ins. *IEEE Trans. Intell. Transp. Syst.* **10**(1), 180–185 (2009)
17. Velickovic, P., Cucurull, G., Casanova, A., Romero, A., Li, P., Bengio, Y.: Graph attention networks. In: International Conference on Learning Representations (2018)
18. Xu, K., Hu, W., Leskovec, J., Jegelka, S.: How powerful are graph neural networks? In: International Conference on Learning Representations (2019)
19. Xu, Y., Piao, Z., Gao, S.: Encoding crowd interaction with deep neural network for pedestrian trajectory prediction. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2018
20. Zhang, L., She, Q., Guo, P.: Stochastic trajectory prediction with social graph network. arXiv preprint [arXiv:1907.10233](https://arxiv.org/abs/1907.10233) (2019)
21. Zhang, P., Ouyang, W., Zhang, P., Xue, J., Zheng, N.: SR-LSTM: state refinement for LSTM towards pedestrian trajectory prediction. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2019