DTG-LSTM: A Delaunay Triangulation Graph for Human Trajectory Prediction in Crowd Scenes

Zhimiao Shi , Yao Xiao

School of Intelligent System Engineering, Sun Yat-Sen University, Shenzhen, Guangdong, China
shizhm3@mail2.sysu.edu.cn, xiaoyao9@mail.sysu.edu.cn,

Abstract

As the crucial elements of complex traffic scenes, the trajectory prediction of pedestrians is valuable for controlling the self-driving cars and supporting the traffic control scheme. The interaction among pedestrians plays a significant role in affecting the pedestrian motion pattern, especially in a crowded situation. Previous trajectory prediction approaches usually ignore the fact that each pedestrian owns a particular topology structure and the structure is changeable as time goes by. To solve the problem, we propose DTG-LSTM, an encoder-decoder model that extracts the dynamic spatio-temporal features of history trajectory information and predicts the trajectories. Considering the geometric properties of the Delaunay triangle which can capture the feature of the topological relationships among pedestrians, an encoder effectively defining pedestrian neighbors with Delaunay triangulation is designed. Meanwhile, a decoder with three strategies (recursion, teacher-forcing, and mixed teacher-forcing) is attempted to get a better performance. As a result, the strengths of the geometric properties and graph convolutional network are combined in our model. The prediction performance using average displacement error and final displacement error is compared with other baselines on the public datasets (ETH and UCY), and our model shows competitive performance.

1 Introduction

Human trajectory prediction plays an essential role in lots of applications, such as autonomous driving and robots [Zhu *et al.*2019]. Due to the complex disturbances in scenarios and diverse behaviors of human beings, it is quite difficult to predict the pedestrians' trajectory precisely. Generally, the pedestrian trajectories in the consecutive snapshots can be considered as time sequence data, and LSTM model shows the competitive performance in time sequence applications [Wiese and Omlin2009, Prakash *et al.*2016, Park and Ahn2018] (e.g., text generation and machine translation). Social LSTM [Alahi *et al.*2016] predicts the trajectories by LSTM net and considers the interactions among pedestrians by designing a social pooling layer. Followed by the idea, many works [Xue *et al.*2018, Zhang *et al.*2019, Chen *et al.*2021, Xu *et al.*2018] design the different LSTM networks to capture the interaction. The consideration of the interaction between pedestrian effectively improve the prediction precision, and the formulation of the pooling and attention mechanism are two important ways for capturing the interactions. In the model of social-LSTM [Alahi *et al.*2016], a pooling layer is added to the Vanilla LSTM to realize the "social" interaction, and the pedestrians share the same cell. The attention mechanism intuitively reflects the crucial timestamp and critical neighbors for predicting future trajectories. To aggregate the information of neighbors, Some approaches deal with interactions by local-spatial attention [Zhang *et al.*2019]. Considering the trajectory of people is not only affected by the neighborhoods. In social-attention model [Vemula *et al.*2018], Anirudh Vemula et al. propose the soft-attention. The relationship of the pedestrians is not measured by the distance. And they extend the attention range to the global. Limited by the realistic scenes, researchers start to put more features into the network. SS-LSTM [Xue *et al.*2018] takes the scenes scale features in the prediction process. In the NEXT model [Liang *et al.*2019], the visual appearance of pedestrian, body skeletons and interaction with the surroundings are extracted as features. LSTM-based modes make a single trajectory for each pedestrian. Obviously, the pedestrians have several path choices, especially in the crowded surrounding. Social GAN introduces the idea of GAN in the prediction tasks. Some works used generator-discriminator architecture to predict the trajectories [Li *et al.*, Kosaraju *et al.*2019].

Recently, the graph-based algorithm achieves outstanding success in the field of node representation and link prediction. In the human trajectory prediction task, the pedestrian and the interactions among the people create a natural topology structure. Inspired by the graph neural network, Abdullah Mohamed et al. [Mohamed *et al.*2020] embed social interaction among pedestrians in the adjacency matrix through a kernel function. Several articles [Huang *et al.*2019, Giuliari *et al.*2020, Shi *et al.*2021, Zhou *et al.*2021, Sadeghian *et al.*2019] combine the novel mechanism (attention mechanism) with the graph. Based on the graph attention network, Huang et al. [Huang *et al.*2019] capture the spatial in-

teraction features by the graph attention mechanism. STAR model [Xia *et al.*2020] captures the spatial interaction features by a transformer-based graph convolution mechanism combined with the transformer mechanism. All the related works display the advantages of the graph in capturing the spatial features. In the real-world scenario, the relationships among pedestrians are changeable. We can define the set of pedestrian snapshots as a dynamic graph. Furthermore, the dynamic graph model relies on the node representation. Dynamic graphs commonly use snapshots of dynamic networks at equal intervals in time series to obtain discrete sequences of network evolution [Li *et al.*2020, Pareja *et al.*2019, Sankar *et al.*2020]. Youngjoo Seo et al. [Seo *et al.*2018] combine convolutional neural network on graphs to find dynamic patterns. And EvolveGCN model [Pareja *et al.*2019] also makes the message aggression by GCN. The above works learn the dynamic characteristics of the graph by feeding the node embedding of snapshots into the recurrent neural network (LSTM or GRU). Till now, the main characteristic of the current topology graph is fully connected and the stationary graph structure, and few works focus on the topological structure and its time-varying properties.

To overcome the limitations, the DTG-LSTM model is formulated by combining the encoder-decoder structure in handling time series data and the Delaunay triangulation geometric features in indicating topologic interactions. Compared with the typical models, our model shows the competitive performance among the models based on graph neural network. The main contributions of this work are as follows: 1. The space is deconstructed with the Delaunay triangle, and it effectively filters the critical neighborhoods especially in crowded situations. To the best of our knowledge, this is the first work combining the geometric properties and graph convolutional network. 2. The decoder of DGT-LSTM combines the teacher-forcing and recursive decoder strategy, and it improves the model adaptability for different types of scenarios.

2 Model

2.1 Problem Defination

Given the observed coordination of pedestrians in the scenario, (x_i^t, y_i^t) donates the coordination of the i^{th} person at the timestamp t. For pedestrian i, the observed trajectory set is $X = \{(x_i^t, y_i^t,) \mid t = 1,\ 2 \ldots T_{ob}\}$. Our goal is to predict the future trajectory coordination $Y = \{(x_i^t, y_i^t) \mid t = T_{ob+1},\ T_{ob+2},\ \ldots T_{ob+p}\}$, where p denotes the number of frames that need to be predicted, T_{ob} is the final observation timestamp.

2.2 Model Description

As discussed above, the existing works suffer from selecting the neighbors who make the impactions. Taking advantage of the geometric property of the Delaunay Triangle, we propose the DTG-LSTM model, which defines the neighbors by Delaunay triangulation. The DTG-LSTM model consists of two main parts: Encoder and Decoder. The relative position coordinates the input of the temporal features, and recurrent GCN extracts the topology features. The encoder extracts the spatio-temporal features by concatenating the relative position and topology embedding. The historical information is concentrated in the hidden states of the encoder. And the decoder is the symmetrical structure of the encoder. The decoder is achieved by three decoder strategies:recursive, teacher-forcing, and mixed-teacher-forcing.

Encoder Process

As a generative model, the encoder extracts the information from the input sequence [Cho *et al.*2014]. To capture the features of multi-timestamps, Long Short-Term Memory (LSTM) is designed to model chronological sequence and their long-range dependencies more precisely than conventional RNNs. And we stack four layers of LSTM to extract the historical information. Figure 1 illustrates the overview of the encoder.

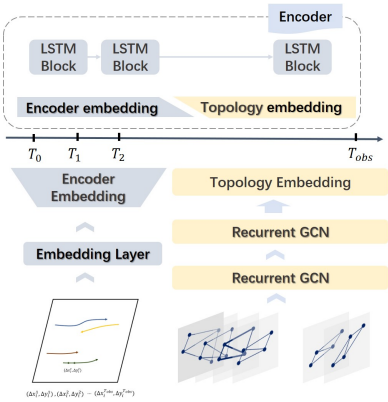


Figure 1: the Encoder Design of the DTG-LSTM Model.

Extraction of Topology Features

The Voronoi diagram of pedestrians shows many valuable properties for motion [Xiao *et al.*2016]. As the dual graph of Voronoi diagram, Delaunay triangulation provides us a specific link of the nodes in the space, which intuitively corresponds the nodes and link with pedestrians and the pedestrian relationships, respectively. Delaunay triangulations also have many geometric properties, such as the closest pair property: Given a point set P, if p_i, p_j are the two closest pair to points, then (p_i, p_j) is an edge of Delaunay triangulations. Corresponding with the real-world scenario, The closest pair property answers the question of who are neighborhoods. Considering some points that are not satisfied with the principle of generating the Delaunay triangulation, we take the fully-connection instead. Compared with the fully-connected topology graph, the Delaunay triangulation split decreases the edges' numbers.

Graph Convolutional Recurrent Networks

Based on the Delaunay triangulation, we get the graph of each frame: $DG_t = (V, E)$. In which, $V = \{P_i^t \mid i \in \{1,\ 2,\ \ldots N\}\}$ and $E = \left\{e_{ij}^t \mid e_{ij} \in DG_{set}\right\}$. Where DG_{set} is the collection of edges in the plot after triangulation split, N is the number of walkers. We and are looking forward to getting the node representation which could reflect the topology structure of the graph.

Given Delaunay graph DG_{set}, for each frame, we get the

normalized Laplacian matrix by:

$$L = I - D^{-\frac{1}{2}} A D^{-\frac{1}{2}} \tag{1}$$

Where A is the adjacent matrix of the graph, and D is the degree matrix. Similar to the Flourier transformation, we define the Fourier Transform of the graph signal as:

$$\hat{f} = U^T f \tag{2}$$

Where U^T is the transpose of the eigendecomposition. $\left(L = U^T \Lambda U\right)$. And the topology structure is represented by matrix Λ. The spectral filter $g_\theta(\Lambda)$ is the convolution kernel. For example, a filter could be defined as:

$$g_\theta(\Lambda) = diag(\theta) \tag{3}$$

In 2016, Defferrard et.al [Defferrard *et al.*2017] fit the convolution filter with a polynomial filter:

$$g_\theta(\Lambda) = \sum_{k=0}^{K-1} \Theta_k \Lambda^k \tag{4}$$

where the parameter $\theta \in R^K$ is a vector of polynomial coefficients. And the calculation of GCN is defined as:

$$\begin{aligned} g_\theta * x_{graph} &= U \sum_{k=0}^{K-1} \Theta_k \Lambda^k U^T x_{graph} \\ &= \sum_{k=0}^{K-1} \Theta_k \left(U \Lambda U^T\right)^k x_{graph} \\ &= \sum_{k=0}^{K-1} \Theta_k L^k x_{graph} \end{aligned} \tag{5}$$

Where x_{graph} are the features of graph. Applied the Chebyshev polynomial, the k-order approximation of g_θ is$g_\theta(\Lambda) \approx \sum_{k=0}^{K} \theta_k^{'} T_k\left(\tilde{\Lambda}\right)$. And $\tilde{\Lambda} = 2 * \frac{\Lambda}{\lambda_{max}} - I$. Where λ_{max} is the maximum eigenvalue.

We leverage the Recurrent GCN network [Seo *et al.*2018] as the node embedding. The idea of Recurrent GCN is feeding the LSTM network with the vector after the aggregation operations of GCN:

$$graph_{embedding} = RGCN\left(x_{graph},\ channel_{size},\ K\right) \tag{6}$$

$$graph_{embedding} = RGCN\left(channel_{size},\ graph_{size}, K\right) \tag{7}$$

Where x_{graph} are the spatial position coordinates, $channel_{size}$ and $graph_{size}$ are the output channel dimension of the GCRN model and graph embedding dimension, respectively. And k is the number of hops on the graph.

the Extraction of Temporal Features

Without other distractions, a pedestrian tends to approach the target with linear trajectories. In our implementation, the relative position is calculated by considering the adjacent timesteps:

$$\left(\Delta x_t^i, \Delta\, y_t^t\right) = \left(x_t^i - x_t^{i-1},\ y_t - y_t^{i-1}\right) \tag{8}$$

Then the relative position is embedded by the embedded layer. And the embedded vector is put into the LSTM encoder as features.

$$e_i^t = \phi\left(\Delta x_i^t,\ \Delta y_i^t,\ W_{ee}\right) \tag{9}$$

Where ϕ is the embedding function parameterized by W_{ee}. For each LSTM block, the encoder embedding is used as input to the LSTM cell.

$$pos_{embedding} = MLP\left(W_{decoder},\ e_i^t\right) \tag{10}$$

Where W_{ee} and $W_{decoder}$ are weight parameters of encoder embedding layers.

Decoder Process

Previous decoder methods can be summarized into two categories: recursive and teacher-forcing. The recursive decoder requires the output of the encoder to acquire as much information as possible from historical sequences to predict the future. Moreover, the error of multi-step prediction is accumulated by recursion. Instead of summing errors from incoming units (possibly erroneous), teacher-forcing solves this problem by sending in all the ground truth in each step. But the teacher-forcing strategy undermines the ability of the multi-step forecasting model. An essential step of our pipeline is to take the mixture of recursive and teacher-forcing. We set a teacher forcing ratio, representing the probability of applying the teacher-forcing strategy. Figure 2, Figure 3, and Figure 4 show the strategies of the decoder.

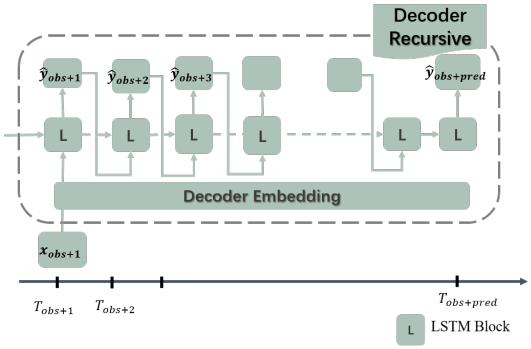


Figure 2: the Recursive Decoder Strategy

With the spatio-temporal information extracted from the decoder, we predict the relative position coordinates at time $t+1$ as following:

$$\left(\Delta \hat{x_i}^t,\ \Delta \hat{y_i}^t\right)^T = LSTM\left(W_{decoder},\ hidden_{states}\right) \tag{11}$$

Where $W_{decoder}$ is the parameter matrix of the decoder embedding layer. And the real coordinate of pedestrian is:

$$\left(\hat{x_i}^{t+1},\ \hat{y_i}^{t+1}\right) = \left(x_i^t + \Delta \hat{x_i}^{t+1},\ y_i^t + \Delta \hat{y_i}^{t+1}\right) \tag{12}$$

3 Dataset and Evaluation Metrics

The model is trained on two public datasets: ETH [Pellegrini *et al.*2009] and UCY [Lerner *et al.*2007]. The two public datasets contain the crowded scenarios, including the

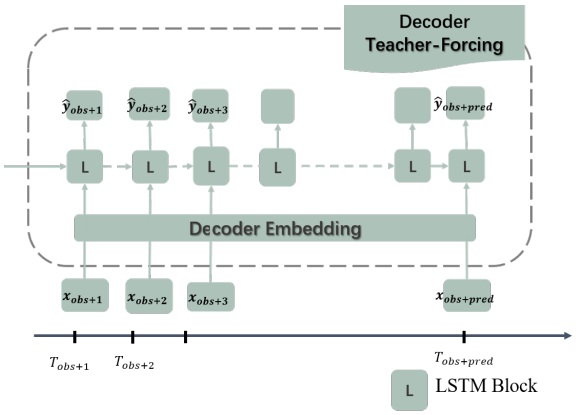


Figure 3: the Teacher-Forcing Decoder Strategy

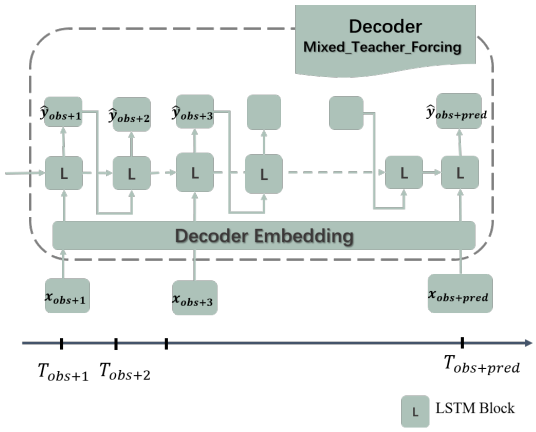


Figure 4: the Mixed-Teacher-Forcing Decoder Strategy

frequent situations: following behavior, group, and collision. Based on the sampling of raw data, we get the pedestrians' coordinates per 0.4 seconds. According to the requirements of the model training, the trajectory of a pedestrian who appears in the 20 consecutive frames is a valid trajectory sequence.In Table1, we count the number of pedestrians that are valid in each dataset. The dataset segmentation is the same as the prior works [Mohamed *et al.*2020]. And the model is trained on 4 datasets and tested on the remaining dataset.

3.1 Evaluation Metrics

In order to have the same evaluation indicators as the previous works, we use two error metrics to report prediction errors:

Average displacement error (ADE): The average Euclidean distance error between ground truth coordinates and the coordinator position that are predicted by the model.

$$ADE = \frac{\sum_{i \in N}^{t \in T_{\text{pred}}} \sqrt{(x_i^t - \widehat{x}_i^t)^2 + (y_i^t - \widehat{y}_i^t)^2}}{N * T_{\text{pred}}} \tag{13}$$

Final displacement error (FDE): the average Euclidean distance error between the final position and the coordination of the final position that are predicted by the model of all the

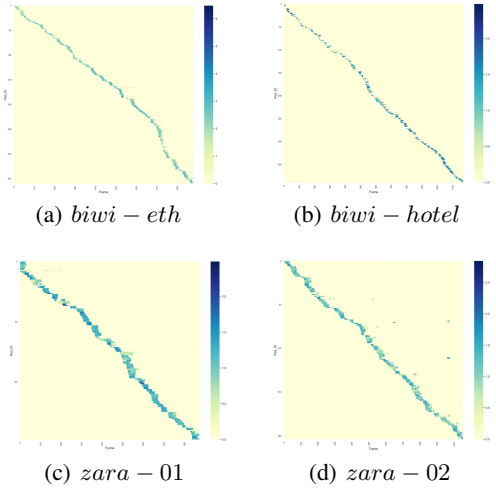


Figure 5: the Frame-Velocity Heat Map of the Datasets I

time steps.

$$FDE = \frac{\sum_{i \in N} \sqrt{\left(x_i^{T_{ob+p}} - \hat{x_i}^{T_{ob+p}}\right)^2 + \left(y_i^{T_{ob+p}} - \hat{y_i}^{T_{ob+p}}\right)^2}}{N} \tag{14}$$

the Spatio-Temporal Analysis of Dateset

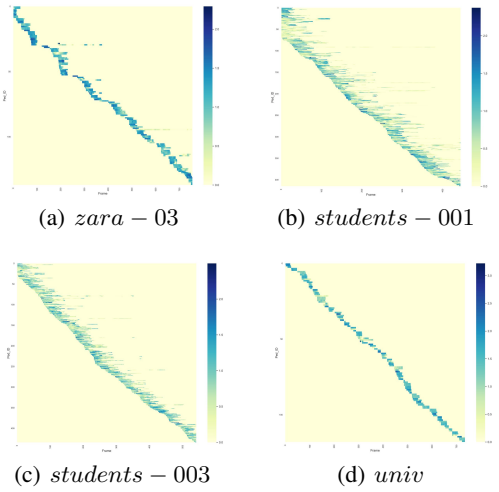
Given the coordinates of pedestrians in every frame, we can get the instantaneous speed by the relative positions (the time interval is 0.4 seconds with the sample operation). Taking biwi-dataset for example, Figure 3 illustrates the Frame-velocity heat map of the dataset. The axis y and x represent the pedestrians and the number of frames, respectively. In addition, the depth of the color reflects the speed of the pedestrians(the deeper, the faster).

Interactions are depicted as the number of pedestrians appearing in the same frame. As shown in Figure 5 and 6, the pedestrians in zara1 and zara2 datasets have more interactions.By further analysis based on visualizing the datasets, the pedestrians of UNIV datasets have the noticeable features that they walk slower than the pedestrian of the other datasets, because the scenarios of the UNIV dataset is more crowded. The regulation reacted by the data visualization is identical to the actual situations, which provides solid data support for analyzing the experiment results.

4 Experiments and Results Analysis

4.1 Implementation Details

We use the embedding dimension of 64 for spatial coordinates before feeding them into the encoder and decoder. And the encoder and decoder have the same architecture: 4 stacked LSTM cells; the hidden size of LSTM size is 128. The decoder strategy is mixed-teacher-forcing, which the teacher-forcing ratio being 0.8. we use a learning rate of 0.001 and Adam for training the model. The model is trained on a single RTX-3060.



(a) $zara-03$ (b) $students-001$

(c) $students-003$ (d) $univ$

Figure 6: the Frame-Velocity Heat Map of the Datasets II

4.2 Baselines

We compare our model with several baselines. Note that some performances in benchmark datasets are saturated, making the gain less pronounced.

LR [Alahi *et al.*2016] a simple linear regression

Social LSTM [Alahi *et al.*2016] The hidden states of is pooled with neighbors. The interaction between the pedestrians reflect on the sharing hidden states.

SR-LSTM [Zhang *et al.*2019] A social-aware information selection mechanism based on encoder-decoder architecture.

Social-Attention [Vemula *et al.*2018] A model that captures the relative importance of neighborhoods when navigating in the crowd.

Social-GAN [Gupta *et al.*2018] A network combines tools from sequence prediction and generative adversarial networks

Sophie [Sadeghian *et al.*2019] A approach blends a social attention mechanism with physical attention mechanism

Social-BiGAT [Kosaraju *et al.*2019] A graph-based generative adversarial network.

Social-STGCNN [Mohamed *et al.*2020] Modeling the interaction as a graph, proposing a kernel function to embed the social interactions.

Table 2 presents the performance comparisons of our method and other typical models: our model well-performed on the ETH and UCY datasets, especially on the ETH dataset.

4.3 Ablation Studies and Analysis

We take the ablation studies on ETH/UCY datasets to verify the validation of model design.

Encoder with Delaunay Triangulation

The design of Delaunay triangulation provides more spatial information to the decoder. We compare the prediction performance of the vanilla LSTM Encoder with DTG-LSTM Encoder.

Decoder Strategy

Inspired by the decoder way of sequence-to-sequence, we implement three decoder strategies: recursive, teacher-forcing, and mixed-teacher-forcing. The most prominent finding to emerge from the analysis is that several datasets perform better with the teacher-forcing decoder. According to the experimental results in Table3, the HOTEL, UNIV, and ZARA1 dataset perform better with the teacher-forcing decoder strategy. ETH dataset and ZARA2 datasets effectively predict with the mixed-teacher-forcing and the recursive decoder strategies, respectively. The teacher-forcing strategy needs the ground truth of the train data to correct the possible error carried by the previous step. Considering the characteristics of interactive trajectories, pedestrians need current information to make decisions more than basing on historical trajectories.

5 Conclusion

In this work, a novel idea combining the geometric properties and graph convolutional network is proposed to extract the dynamic spatio-temporal features of history trajectory information. The relative position features from the stacked LSTM is used to capture the temporal features of the historical trajectories, and the topology features of the pedestrians in the scene is obtained by recurrent GCN architecture. Compared with other fully-connected topology graphs, the critical neighbor can be effectively found and defined, especially in crowded situation. The ADE and FDE indexs show that our model owns a competitive performance with other baselines in the data sets ETH and UCY. Note that the interactions among pedestrians are obviously heterogeneous in real, but the different interactions are still not well illustrated in our model.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant No. 72101276).

Dataset	Pedestrians Number
Biwi-ETH	360
Biwi-HOTEL	389
Crowds-zara02	148
Crowds-zara01	204
Crowds-zara03	137
Students001	415
Students003	434
Univ-examples	118

Table 1: the Number of Pedestrians in the Datesets

	Performance(ADE/FDE)					
Baselines	ETH	HOTEL	UNIV	ZARA1	ZARA2	AVG
LR	1.33/2.94	0.39/0.72	0.82/1.59	0.62/1.21	0.79/1.59	0.79/0.59
Social LSTM	0.86/1.77	0.49/1.00	0.43/1.01	0.38/0.80	0.61/1.24	0.72/1.54
SR-LSTM	0.63/1.25	0.37/0.74	0.41/0.90	**0.32/0.70**	0.51/1.10	0.55/1.14
Social-Attention	1.39/2.39	2.51/2.91	1.25/2.54	1.01/2.17	0.88/1.75	1.41/2.35
Social-GAN	0.81/1.52	0.67/1.37	0.34/0.69	0.42/0.84	0.60/1.26	0.58/1.18
Sophie	0.70/1.43	0.76/1.67	**0.30/0.63**	0.38/0.78	0.54/1.24	0.54/1.15
PIF	0.73/1.65	**0.30/0.59**	0.60/1.27	0.38/0.81	0.31/0.68	0.46/1.00
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
Ours	**0.48/0.89**	0.43/0.90	0.46/0.91	0.40/0.79	0.43/0.84	**0.44/0.86**

Table 2: Camparsion with Baselines Models.

Dataset	Recursive	Teacher_Foring	Mixed
ETH	0.51/0.93	0.49/0.87	0.48/0.89
HOTEL	0.57/1.06	0.43/0.90	0.75/1.78
UNIV	0.60/0.85	0.46/0.91	0.69/1.45
ZARA1	0.61/1.17	0.43/0.84	0.62/1.37
ZARA2	0.43/0.84	0.59/1.17	0.83/1.99

Table 3: Performances of Decoder Strategies.

	ETH	HOTEL	UNIV	ZARA1	ZARA2	AVG
Vanilla Encoder-Decoder	0.74/0.67	0.58/1.17	0.59/1.13	0.75/1.62	0.73/1.46	0.68/1.21
DTG-LSTM (Ours)	0.48/0.89	0.43/0.90	0.46/0.91	0.40/0.79	0.43/0.84	0.44/0.86

Table 4: the Ablation Study of Delaunay Triangulation

References

[Alahi *et al.*, 2016] Alexandre Alahi, Kratarth Goel, Vignesh Ramanathan, Alexandre Robicquet, Li Fei-Fei, and Silvio Savarese. Social LSTM: Human Trajectory Prediction in Crowded Spaces. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 961–971, June 2016.

[Chen *et al.*, 2021] Jinyin Chen, Xueke Wang, and Xuanheng Xu. GC-LSTM: Graph convolution embedded LSTM for dynamic network link prediction. 2021.

[Cho *et al.*, 2014] Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. *arXiv:1406.1078 [cs, stat]*, September 2014.

[Defferrard *et al.*, 2017] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering. *arXiv:1606.09375 [cs, stat]*, February 2017.

[Giuliari *et al.*, 2020] Francesco Giuliari, Irtiza Hasan, Marco Cristani, and Fabio Galasso. Transformer Networks for Trajectory Forecasting. *arXiv:2003.08111 [cs]*, October 2020.

[Gupta *et al.*, 2018] Agrim Gupta, Justin Johnson, Li Fei-Fei, Silvio Savarese, and Alexandre Alahi. Social GAN: Socially Acceptable Trajectories with Generative Adversarial Networks. *arXiv:1803.10892 [cs]*, March 2018.

[Huang *et al.*, 2019] Yingfan Huang, Huikun Bi, Zhaoxin Li, Tianlu Mao, and Zhaoqi Wang. STGAT: Modeling Spatial-Temporal Interactions for Human Trajectory Prediction. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6271–6280, 2019.

[Kosaraju *et al.*, 2019] Vineet Kosaraju, Amir Sadeghian, Roberto Martín-Martín, Ian Reid, S. Hamid Rezatofighi, and Silvio Savarese. Social-BiGAT: Multimodal Trajectory Forecasting using Bicycle-GAN and Graph Attention Networks. *arXiv:1907.03395 [cs]*, July 2019.

[Lerner *et al.*, 2007] Alon Lerner, Yiorgos Chrysanthou, and Dani Lischinski. Crowds by Example. *Computer Graphics Forum*, 26(3):655–664, 2007.

[Li *et al.*,] Jiachen Li, Hengbo Ma, and Masayoshi Tomizuka. Conditional Generative Neural System for Probabilistic Trajectory Prediction.

[Li *et al.*, 2020] Jing Li, Yu Liu, and Lei Zou. DynGCN: A Dynamic Graph Convolutional Network Based on Spatial-Temporal Modeling. In Zhisheng Huang, Wouter Beek, Hua Wang, Rui Zhou, and Yanchun Zhang, editors, *Web Information Systems Engineering – WISE 2020*, pages 83–95, Cham, 2020. Springer International Publishing.

[Liang *et al.*, 2019] Junwei Liang, Lu Jiang, Juan Carlos Niebles, Alexander G. Hauptmann, and Li Fei-Fei. Peeking Into the Future: Predicting Future Person Activities and Locations in Videos. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5718–5727, June 2019.

[Mohamed *et al.*, 2020] Abduallah Mohamed, Kun Qian, Mohamed Elhoseiny, and Christian Claudel. 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)*, pages 14412–14420, Seattle, WA, USA, June 2020. IEEE.

[Pareja *et al.*, 2019] Aldo Pareja, Giacomo Domeniconi, Jie Chen, Tengfei Ma, Toyotaro Suzumura, Hiroki Kanezashi, Tim Kaler, Tao B. Schardl, and Charles E. Leiserson. EvolveGCN: Evolving Graph Convolutional Networks for Dynamic Graphs. *arXiv:1902.10191 [cs, stat]*, November 2019.

[Park and Ahn, 2018] Dongju Park and Chang Wook Ahn. LSTM Encoder-Decoder with Adversarial Network for Text Generation from Keyword. In Jianyong Qiao, Xinchao Zhao, Linqiang Pan, Xingquan Zuo, Xingyi Zhang, Qingfu Zhang, and Shanguo Huang, editors, *Bio-Inspired Computing: Theories and Applications*, Communications in Computer and Information Science, pages 388–396, Singapore, 2018. Springer.

[Pellegrini *et al.*, 2009] S. Pellegrini, A. Ess, K. Schindler, and L. van Gool. You'll never walk alone: Modeling social behavior for multi-target tracking. In *2009 IEEE 12th International Conference on Computer Vision*, pages 261–268, September 2009.

[Prakash *et al.*, 2016] Aaditya Prakash, Sadid A. Hasan, Kathy Lee, Vivek Datla, Ashequl Qadir, Joey Liu, and Oladimeji Farri. Neural Paraphrase Generation with Stacked Residual LSTM Networks. *arXiv:1610.03098 [cs]*, October 2016.

[Sadeghian *et al.*, 2019] Amir Sadeghian, Vineet Kosaraju, Ali Sadeghian, Noriaki Hirose, Hamid Rezatofighi, and Silvio Savarese. 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)*, pages 1349–1358, June 2019.

[Sankar *et al.*, 2020] Aravind Sankar, Yanhong Wu, Liang Gou, Wei Zhang, and Hao Yang. DySAT: Deep Neural Representation Learning on Dynamic Graphs via Self-Attention Networks. In *Proceedings of the 13th International Conference on Web Search and Data Mining*, pages 519–527. Association for Computing Machinery, New York, NY, USA, January 2020.

[Seo *et al.*, 2018] Youngjoo Seo, Michaël Defferrard, Pierre Vandergheynst, and Xavier Bresson. Structured Sequence Modeling with Graph Convolutional Recurrent Networks. In Long Cheng, Andrew Chi Sing Leung, and Seiichi Ozawa, editors, *Neural Information Processing*, Lecture Notes in Computer Science, pages 362–373, Cham, 2018. Springer International Publishing.

[Shi *et al.*, 2021] Liushuai Shi, Le Wang, Chengjiang Long, Sanping Zhou, Mo Zhou, Zhenxing Niu, and Gang Hua. SGCN:Sparse Graph Convolution Network for Pedestrian

Trajectory Prediction. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8990–8999, Nashville, TN, USA, June 2021. IEEE.

[Vemula *et al.*, 2018] Anirudh Vemula, Katharina Muelling, and Jean Oh. Social Attention: Modeling Attention in Human Crowds. May 2018.

[Wiese and Omlin, 2009] Bénard Wiese and Christian Omlin. Credit Card Transactions, Fraud Detection, and Machine Learning: Modelling Time with LSTM Recurrent Neural Networks. In Monica Bianchini, Marco Maggini, Franco Scarselli, and Lakhmi C. Jain, editors, *Innovations in Neural Information Paradigms and Applications*, Studies in Computational Intelligence, pages 231–268. Springer, Berlin, Heidelberg, 2009.

[Xia *et al.*, 2020] Beihao Xia, Conghao Wang, Qinmu Peng, Xinge You, and Dacheng Tao. A Spatial-Temporal Attentive Network with Spatial Continuity for Trajectory Prediction. *arXiv:2003.06107 [cs]*, March 2020.

[Xiao *et al.*, 2016] Yao Xiao, Ziyou Gao, Yunchao Qu, and Xingang Li. A pedestrian flow model considering the impact of local density: Voronoi diagram based heuristics approach. *Transportation Research Part C: Emerging Technologies*, 68:566–580, July 2016.

[Xu *et al.*, 2018] Kaiping Xu, Zheng Qin, Guolong Wang, Kai Huang, Shuxiong Ye, and Huidi Zhang. Collision-Free LSTM for Human Trajectory Prediction. In Klaus Schoeffmann, Thanarat H. Chalidabhongse, Chong Wah Ngo, Supavadee Aramvith, Noel E. O'Connor, Yo-Sung Ho, Moncef Gabbouj, and Ahmed Elgammal, editors, *MultiMedia Modeling*, pages 106–116, Cham, 2018. Springer International Publishing.

[Xue *et al.*, 2018] Hao Xue, Du Q. Huynh, and Mark Reynolds. SS-LSTM: A Hierarchical LSTM Model for Pedestrian Trajectory Prediction. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1186–1194, 2018.

[Zhang *et al.*, 2019] Pu Zhang, Wanli Ouyang, Pengfei Zhang, Jianru Xue, and Nanning Zheng. SR-LSTM: State Refinement for LSTM towards Pedestrian Trajectory Prediction. *arXiv:1903.02793 [cs]*, March 2019.

[Zhou *et al.*, 2021] Yutao Zhou, Huayi Wu, Hongquan Cheng, Kunlun Qi, Kai Hu, Chaogui Kang, and Jie Zheng. Social graph convolutional LSTM for pedestrian trajectory prediction. *IET Intelligent Transport Systems*, 15(3):396–405, 2021.

[Zhu *et al.*, 2019] Yanliang Zhu, Deheng Qian, Dongchun Ren, and Huaxia Xia. StarNet: Pedestrian Trajectory Prediction using Deep Neural Network in Star Topology. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8075–8080, 2019.