Goal-driven Long-Term Trajectory Prediction Supplementary Material

Hung Tran, Vuong Le, Truyen Tran
Applied AI Institute, Deakin University, Geelong, Australia
{tduy,vuong.le,truyen.tran}@deakin.edu.au

1. Introduction

In this supplementary document we include:

- Further implementation details
- Numeric reports on quantitative comparison between GTP and the baselines,
- Additional qualitative results
- A source package containing core method implementation and instruction to reproduce the reported results

2. Implementation details

2.1. Agent-centric representation

We compute the affine transformation from the world coordinate (Cartesian) to the agent-centric coordinate:

$$[v'_1, v'_2, v'_3] = \begin{bmatrix} \phi_{11} & \phi_{12} \\ \phi_{21} & \phi_{21} \end{bmatrix} \cdot [v_1, v_2, v_3] + \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix},$$

where v'_1 is the root $(0, 0)$ and v'_2, v'_3 are the two unit vectors $(-1, 0), (0, 1)$ of the agent-centric coordinate. v_1, v_2, v_3 are the three corresponding vectors in the world coordinate. We choose $v_1 = (x_{t_{obs}}, y_{t_{obs}})$, $v_2 = (x_{t_{obs}} - x_1, y_{t_{obs}} - y_1)$, and v_3 is a perpendicular vector of v_2 at $(x_{t_{obs}}, y_{t_{obs}})$ that has the same length of v_2.

2.2. Network parameters and settings

In this paper, we use GRU in both goal channel and trajectory channel. The dimensions of the hidden state in all GRUs is 16. Similar to [1, 3], we represent q_t as the relative position (or velocity) at time t. We embed q_t to an 8-dimensional vector. For semantic segmentation we use the pretrained model [1] provided in [5]. We use two separate Adam optimizers with the initial learning rate of 0.001 for goal channel and trajectory channel. In each stage in the 3-stage learning process, we train the model with batch size 64 for 150 epochs.

[1] https://github.com/CSAILVision/semantic-segmentation-pytorch

2.3. Destination selection

As mentioned in the paper, the destination selection process includes 5 steps:

Background subtraction and segmentation: For each dataset, we extract the background image B from the provided video using the OpenCV implementation of Apdaptive Mixture of Gaussian [7].

We then segment each background B into semantic areas with the Cascade Segmentation Module [6] trained on ADE20K dataset [5]:

$$S, F = \text{semantic\_parse}(B),$$

where S contains the scores of segmentation and F is the feature map extracted from the penultimate layer of the model. Both tensors have the same spatial size as the image; S has the depth 150 corresponding to categories and features in F has the length of 512.

Border blocks selection We divide a scene into 16×16 blocks. Among these blocks, we consider those at the boundary and the ones next to them as border blocks. These border blocks are demonstrated in Fig. 3 in the paper.

Feature computation For each border block, we compute the semantic feature by average pooling the feature maps F from its pixels, resulting in a 512-dimensional vector.

Border block clustering We then cluster the border blocks at each side of the background scene into regions based on their feature similarity. As big destinations could have negative impacts on the performance of the model, we have a threshold to control the size of the clustered regions. During the clustering process, the border blocks will be assigned to a new region if the size of the previous region has exceeded the threshold. At the end of this process, we have the set of regions potentially be destinations for GTP.

Destination filtering For each region, we compute the probability score of ADE20K classes by average pooling the scores S from its pixels, resulting in a 150-dimensional vector. Then, we select the maximum score of the walkable categories in each region, and we compare it against a threshold to determine the destination candidates. Among 150 classes of ADE20K dataset, we consider four classes: "road", "floor", "grass", "sidewalk" to be walkable in selecting potential destinations for GTP.

3. Detailed Quantitative Results

The detailed comparisons between GTP and other baselines are provided in Table 1. These models are evaluated in two commonly used datasets: ETH [4] and UCY [2].

As described in the main text, the results indicate that current SOTA models only have advantages in short-term 12 time-step prediction. These performances decrease significantly when the prediction length increases.

Meanwhile, GTP has much more stable performance in the far-term prediction, as it outperforms all of the baselines when predicting more than 12 time steps. In fact, the farther the prediction, the wider the gap between GTP and other methods.

4. Qualitative Analysis

We extend the qualitative analysis given in the main paper in Figure 1. We also provide additional visualizations of Trajectory channel's attention weights in Figure 2.

As can be seen in Figure 1, GRU could only predict future trajectories to follow the observed dynamic. In contrast, GTP, by capturing the pedestrians' goals, can generate future trajectories toward the correct destinations. These behaviors of the two models are similar to what was reported in the main paper.

For the utility visualization in Figure 2, it can be seen that the utility scores are usually blunt at the beginning. These scores are later refined as trajectory prediction progresses.

5. Source code

We include the source code, the dataset, and the pretrained models of GTP to reproduce the results reported in the paper. The detail how to execute the code and reproduce the result can be found in the README.md file located in the *code* directory.

References

[1] Agrim Gupta, Justin Johnson, Li Fei-Fei, Silvio Savarese, and Alexandre Alahi. Social gan: Socially acceptable trajectories with generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2255–2264, 2018. 2.2

[2] Alon Lerner, Yiorgos Chrysanthou, and Dani Lischinski. Crowds by example. In *Computer graphics forum*, volume 26, pages 655–664. Wiley Online Library, 2007. 3

[3] 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5725–5734, 2019. 2.2

[4] Stefano Pellegrini, Andreas Ess, Konrad Schindler, and Luc 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. IEEE, 2009. 3

[5] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 2.2, 2.3

[6] Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ade20k dataset. *International Journal on Computer Vision*, 2018. 2.3

[7] Zoran Zivkovic. Improved adaptive gaussian mixture model for background subtraction. In *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, volume 2, pages 28–31. IEEE, 2004. 2.3

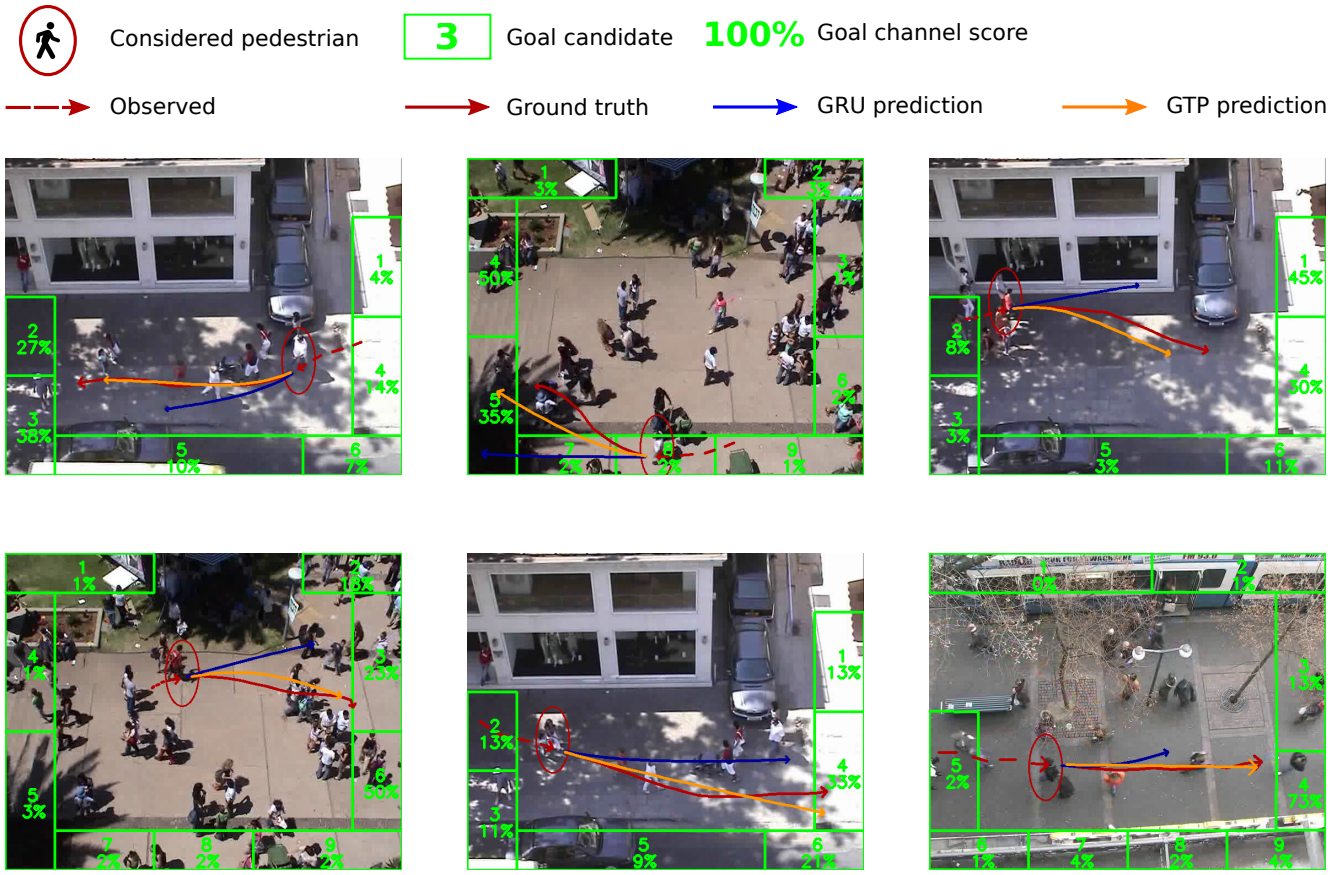

Figure 1. Additional qualitative evaluation of GTP and GRU. Similar to the results shown in the main paper, GRU could only forecast simple moving patterns, while A could capture the goals and generate correct trajectories.

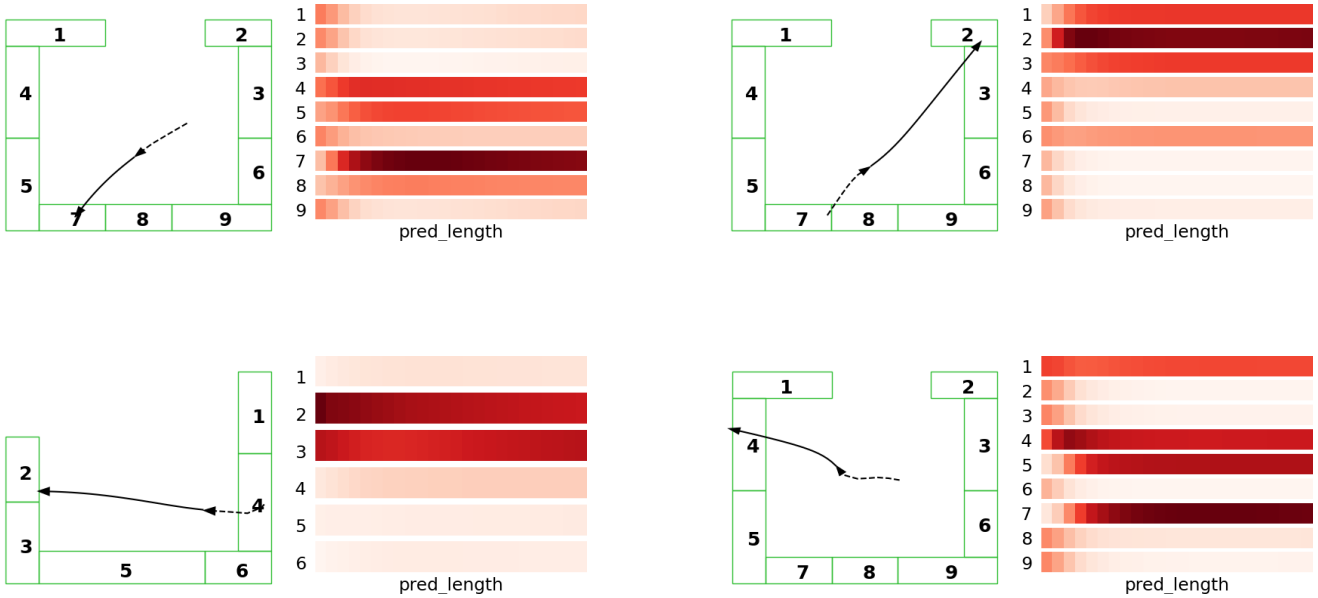

Figure 2. Extended visualization of the utlity score in GTP decoder. The more the pedestrians move, the sharper the utilities. The network gives more scores to the destinations that are in the direction of pedestrians.

	Method	ETH	HOTEL	UNIV	ZARA1	ZARA2	AVG
obs8 - pred12	GRU	0.91 / 1.95	0.63 / 1.33	0.62 / 1.32	0.42 / 0.92	0.32 / 0.69	0.58 / 1.24
	SLSTM	1.09 / 2.35	0.79 / 1.76	0.67 / 1.40	0.47 / 1.00	0.56 / 1.17	0.72 / 1.54
	SGAN	0.87 / 1.62	0.67 / 1.37	0.76/ 1.52	0.35 / 0.68	0.42 / 0.84	0.61 / 1.21
	Next	0.88 / 1.98	0.36 / 0.74	0.62 / 1.32	0.42 / 0.90	0.34 / 0.75	0.52 / 1.14
	SR-LSTM	**0.63 / 1.25**	0.37 / 0.74	**0.41 / 0.90**	**0.32 / 0.70**	0.51 / 1.10	**0.45 / 0.94**
	GTP (Our)	0.82 / 1.68	**0.34 / 0.63**	0.61 / 1.33	0.42 / 0.92	**0.36 / 0.78**	0.51 / 1.07
obs8 - pred16	GRU	0.89 / 2.01	0.62 / 1.28	**0.87 / 1.87**	0.64 / 1.47	**0.42 / 0.93**	0.69 / 1.51
	SLSTM	0.94 / 1.60	0.87 / 1.80	1.09 / 2.33	0.88 / 2.00	0.70 / 1.61	0.9 / 1.87
	SGAN	0.56 / 0.85	0.64 / 1.33	0.98 / 2.03	**0.55 / 1.10**	0.48 / 0.99	0.64 / 1.26
	Next	1.06 / 2.57	0.49 / 1.08	0.95 / 2.1	0.65 / 1.45	0.54 / 1.25	0.79 / 1.74
	SR-LSTM	1.58 / 3.71	0.57 / 1.43	0.95 / 2.11	0.71 / 1.67	0.55 / 1.32	0.87 / 2.05
	GTP (Our)	**0.66 / 1.15**	**0.37 / 0.67**	0.87 / 1.91	0.60 / 1.32	0.44 / 0.97	**0.59 / 1.20**
obs8 - pred20	GRU	0.85 / 1.84	0.64 / 1.3	1.13 / 2.41	0.92 / 2.08	**0.56 / 1.27**	0.82 / 1.78
	SLSTM	1.09 / 2.15	0.92 / 1.83	1.35 / 2.91	1.20 / 2.73	0.87 / 2.00	1.09 / 2.32
	SGAN	0.59 / 0.90	0.94 / 2.08	1.51 / 3.11	0.79 / 1.55	0.63 / 1.34	0.89 / 1.8
	Next	1.14 / 2.78	0.67 / 1.64	1.18 / 2.6	0.93 / 2.1	0.73 / 1.77	0.93 / 2.18
	SR-LSTM	1.13 / 2.64	0.61 / 1.29	1.72 / 4.26	1.62 / 4.56	1.05 / 2.7	1.23 / 3.1
	GTP (Our)	**0.89 / 1.42**	**0.44 / 0.81**	**1.08 / 2.35**	**0.8 / 1.79**	0.57 / 1.29	**0.76 / 1.53**
obs8 - pred24	GRU	0.87 / 1.77	0.64 / 1.29	1.44 / 3.0	1.2 / 2.71	**0.61 / 1.41**	0.95 / 2.0
	SLSTM	1.57 / 3.53	1.05 / 2.09	1.68 / 3.62	1.56 / 3.59	1.09 / 2.45	1.39 / 3.06
	SGAN	0.53 / 0.97	0.55 / 1.10	1.59 / 3.30	1.30 / 2.73	0.77 / 1.68	0.95 / 1.96
	Next	1.05 / 2.2	0.63 / 1.5	1.45 / 3.2	1.33 / 3.07	0.89 / 2.24	1.07 / 2.44
	SR-LSTM	1.48 / 3.31	1.06 / 2.48	2.08 / 5.73	1.53 / 3.65	0.81 / 2.02	1.39 / 3.44
	GTP (Our)	**0.74 / 1.25**	**0.52 / 0.93**	**1.28 / 2.80**	**1.03 / 2.25**	0.64 / 1.47	**0.84 / 1.74**
obs8 - pred28	GRU	0.83 / 1.87	0.68 / 1.34	1.7 / 3.47	1.55 / 3.43	0.62 / 1.49	1.08 / 2.32
	SLSTM	1.55 / 3.70	1.10 / 2.20	2.01 / 4.22	1.93 / 4.31	1.28 / 2.93	1.57 / 3.47
	SGAN	0.74 / 1.45	0.82 / 1.73	1.72 / 3.40	1.46 / 3.09	0.78 / 1.68	1.1 / 2.27
	Next	1.18 / 2.89	0.9 / 2.26	1.73 / 3.77	1.93 / 4.33	1.01 / 2.56	1.35 / 3.15
	SR-LSTM	1.72 / 3.41	0.99 / 2.4	3.2 / 7.88	2.02 / 4.82	0.8 / 2.06	1.75 / 4.11
	GTP (Our)	**0.61 / 1.13**	**0.59 / 1.06**	**1.46 / 3.16**	**1.3 / 2.84**	**0.63 / 1.44**	**0.92 / 1.93**

Table 1. Detailed comparisions (ADE/FDE) between GTP and other baselines. The smaller the nubmers the better the model.