Probabilistic and Multimodal Trajectory Predictions for Autonomous Driving

Philippe Weingertner

CS230 Deep Learning, Stanford University

pweinger@stanford.edu, https://youtu.be/Y4g6-u2Lu7U

1. Abstract

- We study the problem of Trajectory Predictions for Autonomous Driving
- We investigate different architectures: RNN-LSTM variants and Transformer applicability to trajectory predictions
- We propose enhancements with Spatial Attention in addition to Convolutional Social pooling
- We improve results over a state-of-the-art baseline [2]

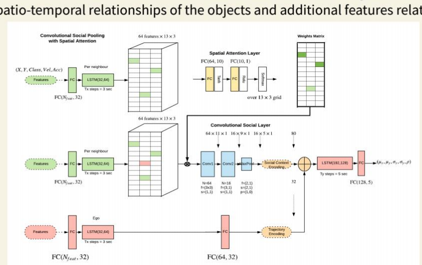
2. Dataset and Features

- NGSIM US Highway 101 dataset (US-101) and Interstate 80 Freeway dataset (I-80)
- The datasets of 90 minutes recording is captured from a bird's-eye view of the highway with a static camera at 10 Hz
- 8.3 millions samples split into 70,10,20 % for the training, development and test set, as used in [2]
- We use only legacy and raw NGSIM features: $(x, y, Vel, Accel, Class)$. Additional Behavioral features were not experimented

3. Methods

3.1 Convolutional Social pooling enhanced with Spatial Attention: CSSA-LSTM(M)

We enhance CS-LSTM(M) [2]: like a human driver we do not focus equally on every neighbors and we learn the best attention weights depending on the spatio-temporal relationships of the objects and additional features related to behavior and shapes.



3.2 Loss Function

We predict a 2D trajectory with a multimodal and probabilistic model: at each time step, a 5D vector corresponding to the parameters of a bivariate Gaussian distribution is derived. For maneuver predictions we use cross-entropy loss functions.

- $f(x,y) = \frac{1}{2\pi\sigma_X\sigma_Y\sqrt{1-\rho^2}} \exp\left(-\frac{1}{2(1-\rho^2)}\left[\frac{(x-\mu_x)^2}{\sigma_X^2} + \frac{(y-\mu_Y)^2}{\sigma_Y^2} - \frac{2\rho(x-\mu_X)(y-\mu_Y)}{\sigma_X\sigma_Y}\right]\right)$
- $\mu = \begin{bmatrix}\mu_X \\ \mu_Y\end{bmatrix}, \mathbf{\Sigma} = \begin{bmatrix}\sigma_X^2 & \rho\,\sigma_X\sigma_Y \\ \rho\,\sigma_X\sigma_Y & \sigma_Y^2\end{bmatrix}$ with -1$\leq \rho \leq$1, σ_X>0, σ_Y>0
- $L_{\text{nll}}(\text{target}=\begin{bmatrix}x \\ y\end{bmatrix}, \text{predicted}=\begin{bmatrix}\mu_X \\ \mu_Y \\ \sigma_X \\ \sigma_Y \\ \rho\end{bmatrix}) = \log\left(\sigma_X\sigma_Y\sqrt{1-\rho^2}\right) + \frac{1}{1-\rho^2}\left[\frac{(x-\mu_x)^2}{\sigma_X^2} + \frac{(y-\mu_Y)^2}{\sigma_Y^2} - \frac{2\rho(x-\mu_X)(y-\mu_Y)}{\sigma_X\sigma_Y}\right]$

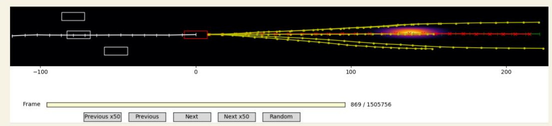
$$\text{Loss} = L_{\text{nll}} + L_{\text{Crossent-lateral}} + L_{\text{Crossent-longitudinal}}$$

4. Experiments and Results

4.1 Experiments

- Teacher forcing results in overfitting for all models; Batch size should be increased as much as possible for Transformer [4]
- For Transformer [3]: with a smaller dataset, we tend to overfit even with dropouts. Finally we use a smaller model with $N_{layers} = 1, d_{model} = 256, d_{feed-forward} = 256, h_{heads} = 4$; lots of proposed optimizations and tricks in [3] are NLP specific
- Seq2seq is 10 times smaller, faster to train (per epoch) and to converge (fewer epochs) than Transformer for similar accuracy
- RNN-LSTM: using a seq2seq architecture, a bidirectional encoder, additional layers, increasing the decoder size and varying the default settings of CS-LSTM(M) does not improve over the baseline [2]
- Spatial attention capturing weighted interactions is more useful than temporal attention (weighting only grids and not grid cells)

4.2 Visualization



The bivariate gaussian is visualized for most probable maneuver at a time horizon of 3 seconds: $\sigma_{\text{longitudinal}} \gg \sigma_{\text{lateral}}$

4.3 RMSE Results on NGSIM dataset

Time (sec)	CV	Deo and Trivedi [2]	Seq2seq	**Transformer**	**CSSA-LSTM(M)**
1	0.73	0.58	0.59	0.52	**0.42**
2	1.78	1.27	1.28	1.23	**1.06**
3	3.13	2.12	2.14	2.17	**1.85**
4	4.78	3.19	3.25	3.23	**2.85**
5	6.68	4.51	4.59	4.70	**4.11**

We improve by enabling additional features processing capabilities with Spatial Attention on top of the Convolutional Social layer

5. Conclusions and Future Work

- We investigated how to apply Transfomer models to trajectory predictions
- We enhanced Convolutional Social pooling with Spatial Attention
- We improved results over a state-of-the-art baseline [2] by 10%
- Future work: experiment in heterogeneous urban environments where Spatial Attention should be even more relevant

6. References

[1] Ernest Cheung. Identifying driver behaviors using trajectory features for vehicle navigation. 2018.
[2] N. Deo and M.M. Trivedi. Convolutional social pooling for vehicle trajectory prediction. CVPR, 2018.
[3] Lukasz Kaiser et al. Attention is all you need. NIPS, 2017.
[4] Martin Popel and Ondrej Bojar. Training tips for the transformer model. 2018.

References

[1] Ernest Cheung. Identifying driver behaviors using trajectory features for vehicle navigation. 2018.

[2] N. Deo and M.M. Trivedi. Convolutional social pooling for vehicle trajectory prediction. *CVPR*, 2018.

[3] Lukasz Kaiser et al. Attention is all you need. *NIPS*, 2017.

[4] Martin Popel and Ondrej Bojar. Training tips for the transformer model. 2018.