Motion Forecasting for Autonomous Vehicles using Argoverse Dataset

Kartik Patath Sapan Santosh Agrawal Rishi Teja Madduri Soumya Srilekha Balijepally

***Abstract*— A better understanding of agents' behaviour in a dynamic traffic environment is required for an efficient modelling and navigation of autonomous vehicles. In this project we plan to address the problem of motion forecasting of traffic actors through experimentation on the Argoverse Motion Forecasting dataset. We attempt to tackle this challenge using generative adversarial networks (GANs) and compare out results with baseline methods of seq-to-seq prediction and social LSTM provided by the Argoverse Challenge.**

***Index Terms*—Behavior Prediction, Autonomous vehicles, GANs, LSTM**

I. Introduction

Autonomous vehicles will soon share roads with the humans, and it is essential for the vehicle to estimate the human driver's intention in order to make better decision. This process of proactively anticipating the traffic actor's future behavior is generally termed as motion forecasting (a.k.a. behavior prediction). That being said, modelling the behavior of a vehicle is a challenging problem [1]. This requires a deep understanding of the subtle traffic scenarios which include interdependence with multiple obstacles (pedestrians and vehicles). Hence, predicting the behaviour of a vehicle also requires modelling the behaviour of the surrounding vehicles/pedestrians. Further more there are challenges such as well structured motion of a vehicle and it's high-inertia which doesn't allow them to instantly change their trajectory. Also there is the problem of multi-modal behaviour of vehicles which means given the fast motion of a vehicle, there may exist more than one possible future behaviour for it.

The Argoroverse motion forecasting dataset [2] is a large-scale collection of vehicle trajectories given by Argo AI. In this work, we use the motion forecasting dataset given by Argo AI for two reasons. First, the dataset includes map data which is critical to the development real world autonomous systems. Second, it captures interesting scenarios such turns at intersections, lane changes and driving with many vehicles nearby. A sample of the argoverse data with map features can be seen in Fig. 1.

A review of various deep-learning based vehicle behavior approaches has been provided by [3]. Which states that Long-Short Term Memory networks(LSTM) are very successful and commonly used to model and learn sequence to sequence mapping using the Encoder-Decoder modules [4]. In recent years, there has been great progress in pedestrian trajectory prediction using datasets such as ETH [5](slpit into ETH and Hotel) and UCY [6](split into ZARA-01, ZARA- 02 and UCY). Some of the most popular works on pedestrian trajectory prediction include [7] [8] [9] [10] [11]. [7] uses a LSTM framework with spatial pooling method to model nearby pedestrians. [8] [11] use a Generative model with LSTM Encoder Decoder setup to approach the trajectory prediction problem and have achieved better results than [7].

All of the aforementioned works focus on pedestrian datasets [6], [5]. But, as mentioned earlier vehicles have a lot more constrained motion which is important for the network to capture. There is also the problem of multi-modality which needs to be addressed in vehicle behaviour modelling. That being said [8] models the behaviour of a pedestrian and generates *socially acceptable* trajectories which account for constrained motion of a person in presence of other pedestrians. [8] also captures the multi-modal behaviour of a pedestrian. This clearly addresses the challenges of behaviour prediction of vehicles which were mentioned before. Therefore, In this work we propose to tackle the general problem of motion forecasting using Generative Adversarial Networks (GANs) [8] which has originally been used to predict the motion of the pedestrians interacting with each other. Furthermore, we compare the results from this implementation with the provided baseline methods in [2].

One interesting addition to the existing framework would be to include map features as a network input along with the trajectories. Map features are important when it comes to modeling the behaviour of a vehicle as they provide subtle details of the environment in which the agent navigates and these are provided by the argoverse dataset.

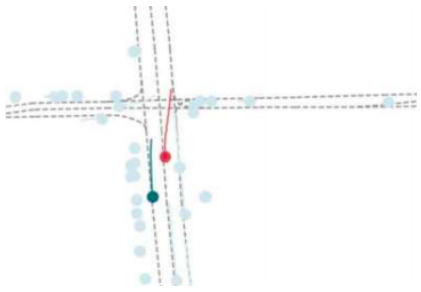

Fig. 1. Sample data from Argoverse Motion Forecasting Dataset [2]. Red - Target Vehicle, Green - Ego Vehicle, Light Blue - Other traffic actors

All authors are from Worcester Polytechnic Institute, (kpatath, ssagrawal, rtmadduri, ssbalijepally)@wpi.edu

A. Research contribution

The key contribution in our work is the data-prepossessing which enabled us to channel vehicle data into the pedestrian trajectory prediction pipeline of Social GAN. We also added the neighbouring agent's trajectories as an input to the network which might help in modelling the interactions between the vehicle and it's neighbourhood. Section-V discusses these in detail.

II. Related Work

Research in trajectory prediction can be majorly classified into types, RNN based, GAN based and Hybrid methods which consist of networks that integrate two or more deep learning architectures.

RNN Models for Trajectory Prediction: The task of sequence prediction has been approached with Recurrent Neural Networks and its varients such as LSTMs and GRUs. LSTMs have been proven useful when analysing spatio-temporal data. In [7] Alahi et al. have proposed Social LSTM which tackels the challenges of Trajectory predcition through a data driven approach. They extend LSTMs for Human trajectory prediction while discussing the shortcomings of naive LSTM implementations in capturing dependencies between multiple correlated sequences. They address this issue by adding a novel "Social" pooling layer which allows the LSTMs to share their hidden state with one another. This results in the Social LSTM automatically learning the interactions between trajectories and accounts for the behaviour of other actors in the scene while predicting a person's path.

Generative Modesl for Trajectory Prediction: Generative models involve a minmax game between a generator and a discriminator. They have mainly been used for tasks that have multiple possible outputs for a given input. In [8] Gupta et al. have presented Social GAN, another pedestrian trajectory prediction network which takes into account the motion of neighbours such as their direction of motion, velocity and the end destination. Their GAN based encoder-decoder architecture addresses the multi-modality problem of trajectory prediction. They use a novel pooling layer which models human-human interaction and a loss function which encourages the network to produce multiple trajectories for the same observed sequence.

Another notable approach that uses adversarial learning for trajectory prediction is the work of Amirian et al. in Social Ways [11]. They use a GAN based LSTM Encoder-Decoder architecture to predict trajectories. In their work they have ignored the L2-loss in training the generator, claiming that it causes mode collapse through faster convergence. They define an attention-based pooling network that borrows from neuroscience and bio-mechanics literature to account for the human-human interaction. To verify the the preservation of multi-modality in trajectory prediction distribution, they present a synthetic dataset of trajectories that can be used to asses the performance of different methods. **Hybrid methods:**

III. Method

Predicting future motion of the vehicles can be a task of learning its underlying physics of motion i.e. kinematics and dynamics of the vehicle. But being driven by a human being, a social element is added to the aforementioned aspects, enabling the vehicle to respect social distance and safety. Thus, to enable more realistic predictions, we take into account the motion of other traffic actors in the environment. In this section, we discuss the GAN based encoder-decoder architecture inspired by [8] which was originally designed for predicting pedestrian motion.

Some terminologies used in the paper are as follows:

- Target vehicle: Vehicle whose trajectory is to be predicted.
- AV vehicle: Autonomous vehicle that collected the data.
- Other agents: traffic actors such as pedestrians, bicycles, other vehicles in the environment.

A. Problem Definition

The motion forecasting task is formulated as: Given the past position trajectories of the traffic actors as $X_i = ({x_i}^t, {y_i}^t)$ for time steps $t = 1, .., T_{obs}$, predict the future trajectory of the Target vehicle ($i = 1$) as $\hat{Y_1} = ({x_1}^t, {y_1}^t)$ for time steps $t = T_{obs+1}, .., T_{pred}$.

B. Socially-Aware GAN

Our model consists of three key components: Generator (G), Pooling Module (PM) and Discriminator (D). G is based on the Encoder-Decoder framework where the hidden states of the Encoder are passed to the Decoder via PM. G takes X_i as the input and outputs the predicted trajectory $\hat{Y_1}$. Discriminator takes the entire trajectory of the target vehicle ($t = 1, .., T_{pred}$) as an input and classifies it as fake/real as shown in the Fig 2.

Generator: First the the location of each traffic actor is embedded using a single layer MLP to get a fixed length vector e_i^t. The embedding is then passed to the LSTM cell of the encoder.

$$\begin{aligned} e_i^t &= \phi(x_i^t, y_i^t; W_{ee}) \\ h_{ei}^t &= LSTM(h_{ei}^{t-1}, e_i^t; W_{enc}) \end{aligned} \tag{1}$$

where, $\phi()$ is that embedding function with ReLU non-linearlity, W_{ee} is the embedding weight. Same LSTM weights W_{enc} are used across all traffic actors in the scene.

Pooling Module: This module does the key job of sharing information across all LSTMS. The Pooling module computes the Euclidean distance of target vehicle from each traffic actor in the scene, concatenated and passed through an MLP to obtain pooled hidden state P.

Traditionally, GANs take as input noise and generate samples. Our goal is to produce future scenarios which are consistent with the past. This is done by conditioning the output of the decoder by initializing the hidden state of the decoder by,

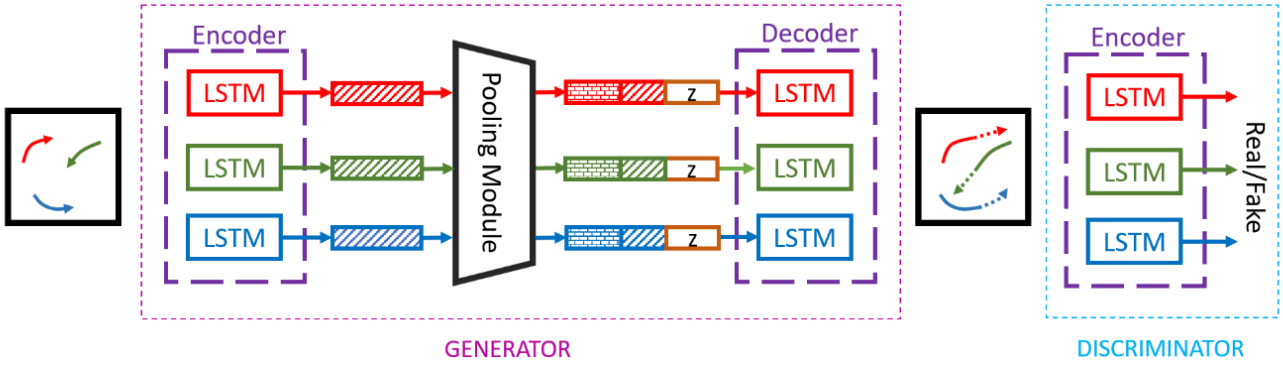

Fig. 2. System overview [8]

$$
\begin{aligned}
c_i^t &= \gamma(P, h_{ei}^t; W_c) \\
h_{di}^t &= [c_i^t, z]
\end{aligned} \tag{2}
$$

To achieve this, the output of the pooling module along with the encoder hidden states are passed through an MLP to get the noise input that can be fed to the decoder. The trajectories are then obtained as:

$$
\begin{aligned}
e_i^t &= \phi(x_i^{t-1}, y_i^{t-1}; W_{ed}) \\
P &= PM(h_{d_1}^{t-1}, h_{d_n}^{t-1}) \\
h_{d_i}^t &= LSTM(\gamma(P, h_{d_i}^{t-1}), e_i^t, W_{decoder}) \\
(\hat{x}_i^t, \hat{y}_i^t) &= \gamma(h_{d_i}^t)
\end{aligned} \tag{3}
$$

Discriminator: The discriminator consists of a separate encoder. Specifically, it takes as input $T_{real} = [X_i, Y_i]$ or $T_{fake} = [X_i, Y_i]$ and classifies them as real/fake. We apply a MLP on the encoder's last hidden state to obtain a classification score. The discriminator will ideally learn subtle social interaction rules and classify trajectories which are not socially acceptable as "fake".

C. Losses

The Generative model G captures the data distribution and a Discriminitive model D that estimates the probability that a sample came from the training data rather than G. The training process is thus similar to a two-player min-ax game with the following objective function,

$$
\begin{aligned}
&\min_G \max_D V(G, D) = \\
&\quad \mathbb{E}_{x\ p_{data}(x)}[logD(x)] + \mathbb{E}_{z\ p(z)}[log(1 - D(G(z)))]
\end{aligned}
$$

For our application, along with the adversarial loss, we used $L2$ loss on the predicted trajectory which measures how far the generated samples are from the actual ground truth.

IV. The Argoverse Dataset

The main inspiration of the project was the Argoverse Motion Forecasting Competition organised by Argo AI in NeurIPS 2019 Workshop. The company launched the Argoverse Dataset to aid the academic community in making advancements in key perception and forecasting tasks for self-driving technology, and to provide resources to explore the impact of high-definition maps on these tasks.

The Argoverse motion forecasting dataset consists of 324,557 five second long sequences of vehicles in interesting scenarios such as 1. vehicles at intersections 2. vehicles taking left or right turns 3. those which are changing adjacent lanes and 4. In dense traffic environments. Each sequence contains the 2D, birds-eye-view centroid of each tracked object sampled at 10hz, as shown in Fig. 1. The dataset can be downloaded from Argoverse Website.

V. Experiments and Results

In this section, we evaluate the performance of our network on the Argoverse Motion forecasting dataset. We divided our sequence of 5 seconds into observation and prediction of 2 seconds and 3 seconds respectively. The Argoverse motion forecasting challenge provides us with an API consisting predefined functionalities to implement the baseline algorithms. However, the API provided falls short of a few functionalities we required for predicting trajectories in a social context. These were later added as a part of data preprocessing.

As a result, majority of our efforts went into data preprocessing to enable us to use the pre-existing Social GANs network for pedestrian dataset on our vehicle trajectory prediction dataset. Some of the key modifications are listed as follows:

- Adding functionality to the Argoverse API to extract neighbour agent's trajectories
- In order to learn the relative motion of the vehicle, we transformed the obtained trajectories to a local reference frame, relative to the agent's position at $t = 0$.
- To ensure consistency in the dataset, we considered only those actors who were present in the environment for a complete sequence of 5 seconds.
- Adding functionalities to the Argoverse API to visualize the predicted trajectories of the agent in the city map.

Evaluation Metrics: We validate our performance with the baseline methods using the following metrics.

1) *Average Displacement Error (ADE)*: Average L2 distance between ground truth and our prediction over all predicted time steps.
2) *Final Displacement Error (FDE):* The distance between the predicted final destination and the true final destination at end of the prediction period T_{pred}.

Baselines: We compared the performance of Social GAN with the baselines provided by the Argoverse.

1) *Linear:* A linear regressor that estimates linear parameters by minimizing the least square error.
2) *LSTM ED*: A simple LSTM Encoder-Decoder framework with no pooling mechanism.
3) *S-LSTM*: The method proposed by Alahiet al. [7]. Each person is modeled via an LSTM with the hidden states being pooled at each time step using the social pooling layer.

The comparative results have been summarized in the Table I.

BASELINE	ADE	FDE
Constant Velocity	3.55	7.89
LSTM ED	2.27	5.19
Social LSTM	1.8	3.89
Social GAN	**0.035**	**0.85**

TABLE I
COMPARISON OF SOCIAL GAN WITH THE BASELINE ALGORITHMS

As seen the the Table I, the Social GAN network has outperformed all the baseline algorithms on the Argoverse dataset. The code to our implementation can be found at our Github repository: https://github.com/sapan-ostic/deep_prediction.git

A. Qualitative Results

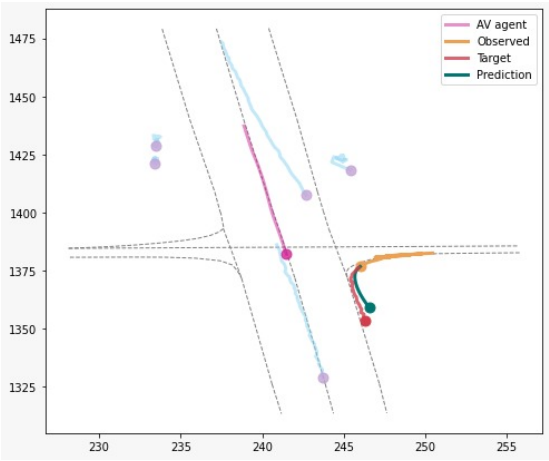

Fig. 3. Left turn predicted by the network correctly.

The qualitative results accurately present the future trajectory of the target vehicle (as seen in Fig. 3 - 5). The trajectory in yellow corresponds to the observed trajectory of the target vehicle. Predicted and target (ground truth) trajectories are in green and red respectively. The trajectory of the AV agent is represented in pink. Motion of other agents is depicted in purple head with blue tail.

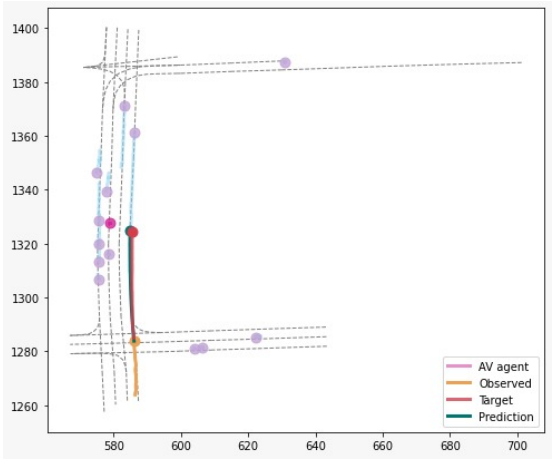

Fig. 4. Position of vehicle at T=5s correctly predicted by the network.

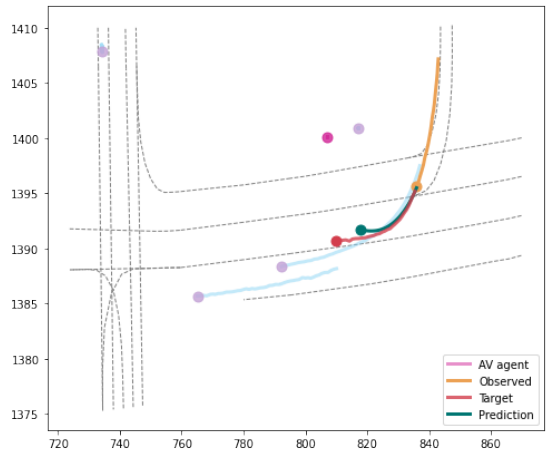

Fig. 5. Right turn predicted by the network correctly.

VI. CONCLUSION AND FUTURE WORK

Implemented Social Generative Adversarial Networks on the Agroverse Motion forecasting dataset to address the issue of motion forecasting. We evaluated this model by computing the predication error considering two metrics i.e Average Displacement error (ADE) and Final Displacement error (FDE). The results have then been compared with the baseline algorithms i.e constant velocity model, LSTM and the social LSTM model that has been provided by the Argoverse Challenge.

In the current implementation, no map information has been provided to the model, resulting in the incorrect predictions of the target vehicle motion as seen in Fig. 3. Thus, the future work involves embedding the map information along with the trajectories of the agents. We also plan to evaluate the performance of the model for nuScenes prediction dataset [12] dataset provided by APTIV. With these improvements we plan to participate in the Argoverse Motion Forecasting Challenge in the upcoming CVPR 2020 conference.

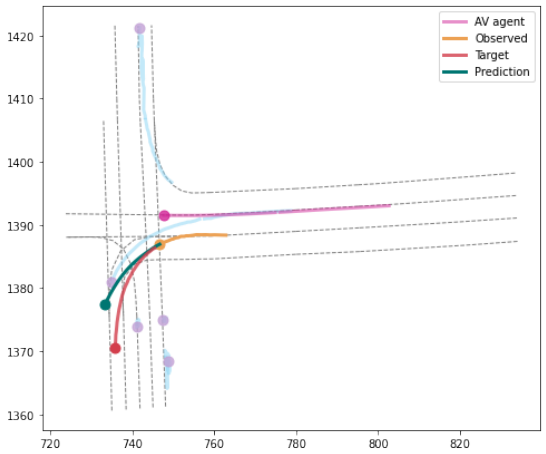

Fig. 6. As no map information is provided to the network, it incorrectly predicts the target vehicle moving out of drivable area.

REFERENCES

[1] C. Badue, R. Guidolini, R. V. Carneiro, P. Azevedo, V. B. Cardoso, A. Forechi, L. Jesus, R. Berriel, T. Paixão, F. Mutz *et al.*, "Self-driving cars: A survey," *arXiv preprint arXiv:1901.04407*, 2019.

[2] M.-F. Chang, J. Lambert, P. Sangkloy, J. Singh, S. Bak, A. Hartnett, D. Wang, P. Carr, S. Lucey, D. Ramanan *et al.*, "Argoverse: 3d tracking and forecasting with rich maps," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 8748–8757.

[3] S. Mozaffari, O. Y. Al-Jarrah, M. Dianati, P. Jennings, and A. Mouzakitis, "Deep learning-based vehicle behaviour prediction for autonomous driving applications: A review," *arXiv preprint arXiv:1912.11676*, 2019.

[4] S. H. Park, B. Kim, C. M. Kang, C. C. Chung, and J. W. Choi, "Sequence-to-sequence prediction of vehicle trajectory via lstm encoder-decoder architecture," in *2018 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2018, pp. 1672–1678.

[5] S. Pellegrini, A. Ess, and L. Van Gool, "Improving data association by joint modeling of pedestrian trajectories and groupings," in *Computer Vision – ECCV 2010*, K. Daniilidis, P. Maragos, and N. Paragios, Eds., 2010, pp. 452–465.

[6] L. Leal-Taixé, M. Fenzi, A. Kuznetsova, B. Rosenhahn, and S. Savarese, "Learning an image-based motion context for multiple people tracking," *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3542–3549, 2014.

[7] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese, "Social lstm: Human trajectory prediction in crowded spaces," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[8] A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, and A. Alahi, "Social gan: Socially acceptable trajectories with generative adversarial networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 2255–2264.

[9] N. Lee, W. Choi, P. Vernaza, C. B. Choy, P. H. Torr, and M. Chandraker, "Desire: Distant future prediction in dynamic scenes with interacting agents," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 336–345.

[10] T. Fernando, S. Denman, S. Sridharan, and C. Fookes, "Soft+ hardwired attention: An lstm framework for human trajectory prediction and abnormal event detection," *Neural networks*, vol. 108, pp. 466–478, 2018.

[11] J. Amirian, J.-B. Hayet, and J. Pettré, "Social ways: Learning multi-modal distributions of pedestrian trajectories with gans," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2019, pp. 0–0.

[12] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuscenes: A multimodal dataset for autonomous driving," *arXiv preprint arXiv:1903.11027*, 2019.