2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

DESTINE: Dynamic Goal Queries with Temporal Transductive Alignment for Trajectory Prediction

Rezaul Karim[1], Soheil Mohamad Alizadeh Shabestary[2], and Amir Rasouli[2]

***Abstract*— Predicting temporally consistent road users' trajectories in a multi-agent setting is a challenging task due to the unknown characteristics of agents and their varying intentions. Besides using semantic map information and modeling interactions, it is important to build an effective mechanism capable of reasoning about behaviors at different levels of granularity.**

To this end, we propose Dynamic goal quErieS with temporal Transductive alIgNmEnt (DESTINE) method. Unlike prior approaches, our approach 1) dynamically predicts agents' goals irrespective of particular road structures, such as lanes, allowing the method to produce a more accurate estimation of destinations; 2) achieves map-compliant predictions by generating future trajectories in a coarse-to-fine fashion, where the coarser predictions at a lower frame rate serve as intermediate goals; and 3) uses an attention module designed to temporally align predicted trajectories via a masked attention operation.

Using the common Argoverse benchmark dataset, we show that our method achieves state-of-the-art performance on various metrics, and further investigate the contributions of proposed modules via comprehensive ablation studies.

I. INTRODUCTION

A key challenge in trajectory forecasting in the context of autonomous driving is modelling latent factors, such as intentions of road users and their behavior while interacting with others. Existing approaches resort to explicit prediction of intentions in the form of multiple probable goals or destinations at the end of prediction horizon [20], [31], [52], [63]. For instance, heuristic methods rely on the scene layout or the vehicles' dynamics to estimate goals [7], [63] and learning-based methods rely on the data distribution [20], [31]. However, given their reliance on static context or past observations for goal prediction, these methods lack the ability to perform well in dynamically evolving scenarios where the intentions of the agents may change or scenarios that have not been observed in the training data, i.e. out-of-distribution scenarios.

To adapt to changes in the scenes, dynamic architectures [25], [57], [65] are widely used in computer vision domain. Rather than directly estimating the model parameters, these techniques learn a series of filter kernels conditioned on the input. During inference time, the behavior of the kernels adaptively changes allowing the model to adjust its processing to the current circumstances [21]. Motivated by these approaches, we propose a dynamic goal predictor module designed based a transformer-based architecture to generate dynamic goal queries during inference to estimate target locations.

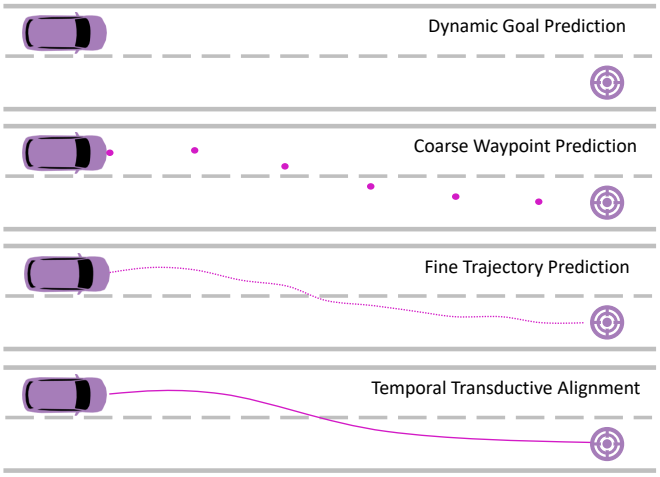

Fig. 1: An overview of the operation procedure of the propose model. Our approach begins by dynamically predicting goal locations, followed by coarse waypoint prediction to define the intermediate points. Next, the model produces fine trajectories which are then fed into the temporal transductive alignment module for final refinement.

Another challenge in prediction is to generate admissible trajectories that comply with road structure as well as dynamical constraints. Methods, such as autoregressive inference [1], [42], heuristic reasoning [47], observation reconstruction [34], [60], and scene graph consistency computation [33] are commonly used to generate admissible trajectories. However, these methods add significant computation overhead, which is undesirable in time-critical applications such as autonomous driving. To address this shortcoming, we propose a simple yet efficient temporal alignment mechanism that enforces consistency over the predicted trajectories based on cause-effect relationships.

In this paper, we introduce the Dynamic goal quErieS with temporal Transductive alIgNmEnt (DESTINE) model, which benefits from an attention-based architecture that generates dynamic queries to estimate target locations as a proxy to latent intents. In addition, our approach generates road structure compliant trajectories using a coarse-to-fine prediction scheme and a temporal transductive alignment (TTA) mechanism. The coarse predictions, generated at a lower sampling rate, serve as intermediate goals to improve fine predictions while the TTA module aligns trajectory points across time. We conduct extensive empirical evaluations on the Argoverse [8] dataset and show that our method achieves state-of-the-art performance on various metrics. We further demonstrate the effectiveness of the proposed modules via

[1] EECS at York University. Work done while at Huawei. karimr31@yorku.ca
[2]Noah's Ark Laboratory, Huawei, Canada. soheil.shabestary@huawei.com, amir.rasouli@huawei.com

ablation studies. In summary **contributions** of our work are as follows:

- We present a novel dynamic architecture for goal prediction that adaptively estimates road users' targets.
- We propose a novel temporal transductive model that aligns predicted trajectory points to increase their compliance to the scene layout.
- We conduct extensive empirical evaluations and highlight the role of novel components using ablation studies.

II. Related Works

Intention Estimation. There is a large body of work dedicated to trajectory prediction for autonomous driving, often specialized on pedestrians [3], [33], [43], [44], [49] and vehicles [27], [35], [53], [66]. Some of the main topics of interest in this domain include, scene representation [1], [6], [10], [11], [18], [19], [36], [41], [42], [45], interaction modelling [12], [19], [22], [23], [29], [40]–[42], [46], [54], [56], [59], [61], [64], [66], latent intention modelling [1], [7], [16], [20], [30], [31], [47], [50], [52], [55], [58], [62], [63], and road structure compliance enhancement [1], [32]–[34], [42], [47], [60].

One of the key challenges in trajectory prediction is modelling future uncertainty stemming from underlying agents' intentions which are not readily foreseeable. Some methods address this challenge by directly learning the distributions of trajectories over latent representations [31], [47], [58] often conditioned on the future states of the agents in the training phase. These methods, however, are prone to mode collapse problem [6], [7] and can potentially become intractable as the space of possibilities grows. Alternatively, anchor-based approaches attempt to learn the space of possibilities, which is highly dependent on the quality of the hand-crafted anchors [7], [16], [30], [50]. Heuristic based memory models estimate intentions from a memory database at inference time but their performance is limited when dealing with out-of-distribution scenarios [55].

Another category of methods estimate goals or potential target locations as a proxy to the intentions of agents [20], [31], [52], [62], [63]. Goals are predicted using cues, such as lane centerlines [62], [63], area heatmap on map [31], or densely sampled drivable areas [20]. Given the reliance of these methods on static architectures, their effectiveness is hindered when exposed to out-of-distribution scenarios. We address this issue by adopting a dynamic architecture for goal prediction inspired by [21], [25], [57], [65]. Using a dynamic approach a subset of model parameters are adjusted during inference time allowing the model to adapt to the scenarios that were not previously seen during training. Given the success of this design approach in different applications of computer vision, such as classification [57], object detection [5], [13], and segmentation [14], [17], [26], we present an attention-based architecture that adapts the model's parameters to estimate targets using dynamically learned goal queries leading to better generalization to out-of-distribution scenarios.

Compliant trajectory prediction. To be reliable, generated trajectories should be compliant to (or consistent with) road structure and dynamical constraints. Existing models achieve this by observation reconstruction [34], [60], scene graph consistency computation [33], or heuristics methods [47]. Alternatively, compliancy can be achieved by autoregressive inference [1], [32], [42]. However, besides the exposure bias problem [48], autoregressive processing is computationally expensive making it a less desirable choice for time-sensitive applications, such as autonomous driving. Another line of work uses the coarse-to-fine approach in which intermediate goals (or waypoints) at a lower sampling rate are predicted as intermediate steps to enforce compliancy [31], [39], [52]. We follow a similar scheme and additionally propose a temporal transductive alignment (TTA) module which learns to align generated trajectories using a masking operation. TTA is computationally efficient and operates on final generated trajectories allowing it to be applied to many existing prediction approaches.

III. Proposed Approach

A. Problem Formulation

In our formulation, we use a continuous space discrete time sample assumption. A traffic scene includes agents and map information. Agents' information consists of their past observed states (coordinates and headings), $S_{obs} = \{s_a^t : t \in T_{obs}, a \in A\}$, where A is the set of all agents in the scene, and $T_{obs} = \{-t_o, ..., 0\}$ are the observation time steps. The map information M includes the geometric and semantic attributes of map component, such as lanes, regions of intersections, and signals. We use a vectorized representation for the entities in each scene [18]. Our model outputs the future states of the agents $S_{pred} = \{s_a^{t,k} | S_{obs}, M : t \in T_{pred}, a \in A, k \in K\}$, which are the coordinates and headings of the agents for K different modes. Here, $T_{pred} = \{1, ..., t_p\}$ are future time steps, $K = \{1, ..., k\}$ are different modes of predicted trajectories, and $s_a^{t,k}$ is the future state of agent a, at time step t for mode k. Each predicted trajectory is associated with a probability score $P = \{p_a^k : a \in A, k \in K\}$ where $\sum_{k \in K} p_a^k = 1$.

B. Model Overview

An overview of our proposed method, DESTINE is depicted in Figure 2. Overall, there are four core modules: **context encoding** where agents' dynamics along with high-dimensional map information are processed to produce a context representation; **dynamic goal prediction**, which receives the context encoding as input and learns goal queries to generate potential goal locations at time t_p for the agents; **Coarse-to-fine trajectory prediction**, which relies on predicted goals and context encoding to generate trajectories; And **temporal transductive alignment** module that refines the generated trajectories using a masked attention operation.

C. Context Encoding

The context encoding is based on the model in [66], in which the translation invariant representation of agents'

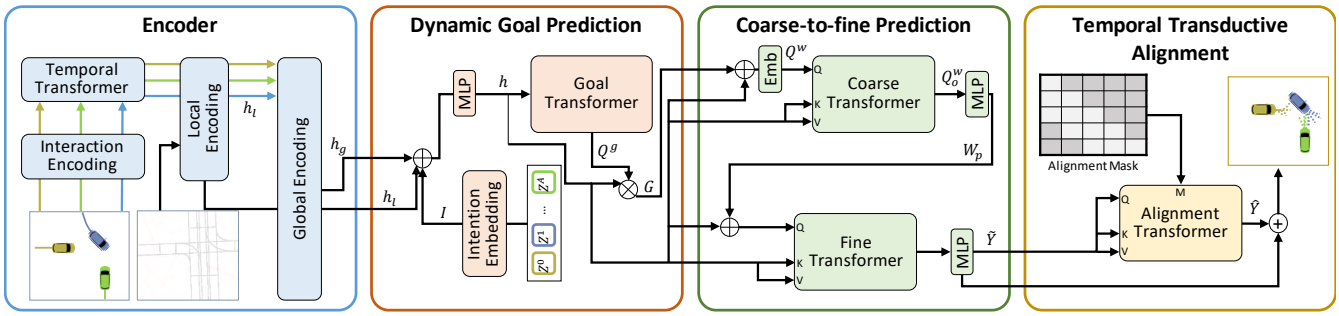

Fig. 2: An overview of the proposed approach. The encoder models the interactions between the agents, and agents and the road. Next, a goal is generated that in conjunction with encodings serves as the query to the coarse trajectory generator, the output of which is used to condition the fine trajectory prediction. At the end, the TTA module temporally aligns the fine trajectories resulting in the final predictions.

attributes and the scene are generated by converting the coordinate reference frame to the center of agent at time $t = 0$. The interactions between agents, and agents and the environment are modelled at two different levels. At the *local* level, patches of radius, $r(= 50m)$, centered at each agent are extracted. At each time step, the patches are processes using a self-attention layer to capture spatial relationships. The output of these patches are aggregated by concatenation and fed into a temporal transformer to model the temporal relations. The output of the transformer, in addition to lane information, are fed into an additional attention layer that captures the local lanes and agents relationships. This process produces a local agent a representation $h_{a,l}$. At the *global* level, a message passing operation followed by a spatial attention is used to model the interactions between local representations. This operation produces a global agent representation $h_{a,g}$.

D. Dynamic Goal Predictor

We define goal as the final future position of predicted trajectory for each mode k, $s_a^{t_p,k}$. If estimated accurately, goals can help improve the compatibility of predictions to the road structure, i.e. result in admissible predictions that do not extend beyond the road boundaries [63].

To enhance diversity and prevent mode collapse, we use an intention embedding layer to generate input for goal prediction. The layer is based on a discrete set of mode representations, where a mode is a concrete instantiation of an intention [1], [47]. In particular, we use intention, z^a, for agent, a, with K one-hot vectors of length K representing each mode. This $Z = \{z^a : a \in A\}$ is then projected to C dimensional embedding space by a linear layer, added to a learnable positional encoding and finally concatenated with the index of the agents to get intention embedding representation for all agents in the scene I. Together, discrete intention embedding, $I = \{I_a : a \in A\}$, local features,$h_l = \{h_{a,l} : a \in A\}$, and global features $h_g = \{h_{a,g} : a \in A\}$ are concatenated to form the final feature encoding, h.

To enhance the generalizability of the goal predictor to out-of-distribution scenarios, inspired by dynamic architectures [5], [25], [26], [65], the goal transformer is designed to generate dynamic filters in the form of goal queries to estimate a set of multimodal goals G from the observation encoding h. In Section IV-E, We highlight the motivation behind this design by comparing our learned dynamic goal queries to the non-dynamic learned goal predictor approaches similar to [20], [63] that use linear or convolutional filters to predict goals from the encoded observation features.

Traditionally, non-dynamic goal prediction models can be represented as $\mathsf{G} = \mathcal{F}^{\int}(\mathsf{W}^{\mathsf{s}}, \mathsf{h})$ where $\mathcal{F}^{\int}$ is the goal predictor function, W^{s} is the model parameters, and h is the embedding from the encoder [31], [63]. The parameter W^{s} is optimized during training from the entire training set and does not adapt during inference. Conversely, in the proposed dynamic query-based goal predictor, the operations are $\mathsf{Q}^{\mathsf{g}} = \mathcal{F}^{d}(\mathsf{W}^{\mathsf{d}}, \mathsf{h})$ and $\mathsf{G} = \mathcal{F}^{q}(\mathsf{Q}^{\mathsf{g}}, \mathsf{h})$. Here, Q^{g}, W^{d}, h and G corresponding to goal queries, goal transformer parameters, input features, and predicted goals, respectively. The function $\mathcal{F}^{d}$ is the goal transformer function that generates the goal query Q^{g} and $\mathcal{F}^{q}$ is the goal prediction function. $\mathcal{F}^{q}$ is analogous to the operation of a linear layer in the multi layer perceptron (MLP) which is a matrix multiplication with Q^{g} as a dynamic filer and h as an input feature. In our model, goal transformer parameter W^{d} is optimized during training to generate suitable queries Q^{g} to estimate goals G. Since the goal queries are analogous to dynamically generated weights or filters of a linear layer, the goal transformer takes a single input h to be used as keys and values, while the queries are generated from a random initialization conditioned on input h. In this context, we use the term *dynamic* because the parameters Q^{g} are generated at the inference time by the goal transformer. This formulation makes the goal predictor more adaptive to a given scene as the transformer learns how to focus on distinctive features generated by the network at the inference time [2].

The goal transformer uses l layers of multi-head attention with the operation of the i^{th} layer given by,

$$\begin{aligned} \mathsf{Q}_i^g &= \mathcal{M}_s(\mathsf{Q}_{i-1}^g + \mathsf{P}_q, \mathsf{Q}_{i-1}^g + \mathsf{P}_q, \mathsf{Q}_{i-1}^g) \\ \mathsf{Q}_i^g &= \mathcal{M}_c(\mathsf{Q}_i^g + \mathsf{P}_q, h + \mathsf{P}_h, h) \\ \mathsf{Q}_i^g &= \mathcal{L}(\mathsf{Q}_i^g) \end{aligned} \tag{1}$$

where $\mathcal{M}_s$, $\mathcal{M}_c$, and $\mathcal{L}$ are self-attention, cross-attention [51], and an MLP respectively. Q_i^g, P_q, P_h are the goal query at i^{th} layer, query position embedding, and feature position embedding, respectively. Q_0^g begins with

a random initialization and Q_l^g is the final goal query, Q^g. Goals G consist of $B = 5$ dimensional vector representing the mean and standard deviation of a Gaussian distribution of the 2D coordinates and the confidence of the given mode.

E. Coarse-to-fine Trajectory Prediction

We employ a multi-granular prediction scheme for improved map compliancy in which we initially predict coarse trajectories with a lower sampling rate (e.g., 1 Hz) resulting in an output trajectory with sparser set of points. These points are often referred to as waypoints [39], or intermediate or short term goals [31]. Given that for a specific start and end point an intermediate point in a short distance is unimodal [63], we predict a single coarse trajectory for each goal. Our model predicts the intermediate points conditioned on both the goal and intention embeddings and outputs predicted points as well as a new estimate of the goal. This allows the module to further refine the initially estimated goal.

The coarse trajectory predictor relies on the transformer decoder formulation [51], details of which are omitted for brevity. The predicted goal G and the feature encoding h are concatenated to serve as the query input to the transformer decoder while h serves as keys and values. Similar to the goal transformer, the inputs are processed using a self-attention, a cross-attention, and an MLP to generate waypoints' features followed by another MLP network to generate the coarse trajectory output $\mathsf{W}_p \in \{R^{N\times K\times T_{wp}\times B}\}$. Here, T_{wp} is the number of waypoint samples. Next, the fine trajectory prediction module produces trajectories at the desired higher sampling rate (e.g. 10 Hz). The architecture of the fine predictor is similar to the coarse one where h and W_p are concatenated and used as the query input to the transformer decoder while h is used as keys and values. The fine predictor estimates final sampled trajectories $\tilde{\mathsf{Y}} \in \{R^{N\times K\times T_{pred}\times B}\}$.

F. Temporal Transductive Alignment

Temporal transductive alignment (TTA) module is designed to enforce temporal cause-effect relationship over the time steps in a non-autoregressive manner. Our design is motivated by the fact that real life driving trajectories are smooth over short time horizon due to the physical constraints of maneuvering vehicles. This means that the short-term trajectories should follow a consistent path without any apparent oscillating pattern. The proposed method realigns the predicted trajectories to achieve better consistency across different time-steps over a short time horizon and consequently achieves better road structure compliancy.

TTA enforces temporal consistency within a short term temporal window by using a masked self-attention mechanism that is applied over the temporal dimension of the predicted trajectory $\tilde{\mathsf{Y}}$ using an analytically designed mask $\mathsf{M_t} \in [0,1]^{T_{pred}\times T_{pred}}$, producing the final predictions $\hat{\mathsf{Y}}$. We define a temporal window tw over which we want to enforce temporal consistency. In this process, the design of the attention mask is important. Through empirical evaluations, we identified a windowed square-subsequent mask to be the most effective in this context. The operation for a single layer of TTA can be summarized as follows:

$$\begin{aligned} \mathsf{Y}' &= \tilde{\mathsf{Y}}_{\mathsf{T}\times(4\times\mathsf{N})\times\mathsf{K}} + \mathsf{P_t} \\ \mathcal{T}(\mathsf{Y}', \mathsf{M_t}) &= \mathsf{S}\left(\frac{1}{\sqrt{d}}\mathsf{Y}'\mathsf{W}_h^q(\mathsf{Y}'\mathsf{W}_h^k)^\top + \mathsf{M_t}\right)\tilde{\mathsf{Y}}\mathsf{W}_h^v \end{aligned} \tag{2}$$

where S is the Softmax operation and $\mathsf{P_t}$ is a learnable position embedding.

G. Learning Objective

Our training loss is the combination of three objective functions given by, $L = \alpha L_g + \gamma L_{wp} + \beta L_{\hat{\mathsf{Y}}}$, where L_g, L_{wp}, and $L_{\hat{\mathsf{Y}}}$ are goal, waypoints (coarse predictions), and final trajectory losses, respectively. Coefficients α, γ, and β are mixing weights which are set empirically. We compute the goal loss L_g by the negative log likelihood (NLL) and Huber loss with equal weights and only use Huber loss for L_{wp}. The final trajectory loss $L_{\hat{\mathsf{Y}}}$ computes negative log likelihood (NLL), Huber loss with equal weights, and an additional classification loss using cross entropy for the confidence score. In all cases, we used only the best trajectory in terms of Final Displacement Errors (FDE) to compute the NLL and Huber loss.

IV. Experiments

A. Implementation Details

The transformers used in our model all have 4 layers with 8 heads and embedding dimensions are set to 128. We chose 1Hz sampling rate for coarse prediction and the standard 10Hz sampling for fine-grained predictions. We set the default masked window horizon $tw = 2s$ in TTA. For training, We used teacher forcing scheme where ground truth end-points (goals) are used as input to the waypoint decoder for the first 60 epochs and then we used the predicted goals to train the model for another 20 epochs. The batch size was set to 32, initial learning rate to $5e^{-4}$ with a cosine annealing learning rate scheduler [37], and weight decay to $1e^{-4}$ using Adam optimizer [38].

B. Benchmarks

Dataset. The existing datasets for vehicle trajectory prediction have significant diversity in data formats, annotation and evaluation protocols. For this reason it is a common practice to evaluate methods on a single large dataset [24], [35], [36], [66].

Following this approach and for a fair comparison with closely related works, we conduct our evaluation on the large scale Argoverse motion forecasting benchmark dataset [8] on both validation and test sets.

Metrics. As in the past arts [19], [41], [66], we use the standard evaluation metrics, namely minimum Average Displacement Error ($minADE$), minimum Final Displacement Error ($minFDE$), and Miss Rate (MR).

Although distance-based metrics capture the accuracy of predictions with respect to the ground truth, they do not highlight the admissibility or compliancy to the road structure. For this purpose, we report on two additional metrics, namely

TABLE I: Evaluation results on the Argoverse validation set for K=6. For all metrics lower values are better.

Models	Venue	minADE	minFDE	MR	HOR	SOR
TNT[63]	CoRL'20	0.95	1.73	0.21	3.00	0.15
LaneGCN[35]	ECCV'20	0.71	1.08	0.10	2.55	0.12
LaPred[27]	CVPR'21	0.71	1.44	-	-	-
MMTrans.[24]	ICRA'22	0.71	1.08	0.10	3.02	0.15
AutoBot[19]	ICLR'22	0.73	1.10	0.12	-	-
HiVT[66]	CVPR'22	0.66	0.97	0.09	2.36	0.11
HLS-MM[9]	ECCV'22	0.65	1.24	-	-	-
DESTINE (Ours)		**0.64**	**0.90**	**0.08**	**2.05**	**0.10**

TABLE II: Evaluation results on the Argoverse test set, for K=1 and 6. For all metrics smaller value is better.

Models	Venue	K=1			K=6		
		minADE	minFDE	MR	minADE	minFDE	MR
TNT[63]	CoRL'20	2.17	4.96	0.71	0.98	1.45	0.17
LaneGCN[35]	ECCV'20	1.70	3.76	0.59	0.87	1.36	0.16
mmTrans.[36]	CVPR'21	1.74	4.00	0.60	0.84	1.34	0.15
DenseTNT[20]	ICCV'21	1.68	3.63	0.58	0.88	1.28	0.13
MMTrans.[24]	ICRA'22	1.74	3.90	0.60	0.84	1.29	0.14
Scene Trans.[41]	ICLR'22	1.81	4.05	0.59	0.80	1.23	0.13
AutoBot[19]	ICLR'22	1.84	4.12	0.63	0.89	1.41	0.16
HiVT[66]	CVPR'22	1.60	3.53	0.55	**0.77**	1.17	0.13
LTP[53]	CVPR'22	1.62	3.55	0.56	0.83	1.30	0.15
DESTINE (Ours)		**1.59**	**3.49**	**0.54**	**0.77**	**1.15**	**0.12**

Hard Off-Road (HOR) and Soft Off-Road (SOR) as proposed in [4]. The former calculates the percentage of predictions that at least have one predicted point of the trajectory off-road and the latter measures the percentage of all off-road points over all predicted points averaged over all scenarios. Unless otherwise mentioned, all evaluations are on $K = 6$. For all metrics, lower values are better.

Models. We selected state-of-the-art models that have officially been evaluated on the Argoverse dataset. On validation set we report the results on TNT [63], LaneGCN [35], LaPred [27], MMTrans. [24], AutoBot [19], HiVT[66], and HLS-MM [9]. To compute the new metrics, we used the official code released by the authors except for TNT[1]. For the test set, in addition to the previous models that have test benchmark results, we report on mmTransformer [36], DenseTNT [20], Scene Transformer [41], and LTP [53].

C. Comparison to State-of-the-art

We begin our experiments by evaluating the proposed method, DESTINE, against state-of-the-art trajectory prediction algorithms. The results of the experiments on the validation set are shown in Table I, in which we can see that our approach achieves the best performance on all metrics. On distance metrics, more notable improvement is achieved on FDE (by 7%) and MR (by 11%) highlighting the importance of effective goal selection and refinement. More improvements are achieved on admissibility metrics (by up to 13% on HOR), showing that the trajectory samples generated by our model are generally more compliant with the road structure. A similar pattern of improvements is achieved on test set as shown in Table II. Once again the benefit of the proposed method is more apparent on FDE and MR metrics where improvement of up to 8% is achieved. The comparison between 1 and 6 mode results shows that our model is not only better at generating good samples but also at selecting the most confident one. In terms of accuracy-efficiency trade-off, we observe 6% improvement in FDE with only 25ms increase (without any optimization) in the runtime for $K = 6$ compared to HiVT [66] (94ms vs 69ms). Such a increase in the computation time is mainly due to the use of the TTA module. However, it can be further minimized by integrating transformer optimization methods, such as [15], [28].

[1]Implementation is from `https://github.com/Henryliu/TNT-Trajectory-Prediction`

TABLE III: Ablation studies on the components of our model.

Trajectory	Goal	Waypoints	TTA	minDE@1s	minDE@2s	minDE@3s	MR
✓	-	-	-	0.53	0.84	0.96	0.090
✓	✓	-	-	0.53	0.81	0.94	0.088
✓	✓	✓	-	0.52	0.79	0.93	0.084
✓	✓	✓	✓	**0.52**	**0.77**	**0.90**	**0.081**

D. Qualitative Results

Examples of our model's performance are depicted in Figure 3 showing the effectiveness of DESTINE in capturing different modes of driving. Of particular interest is the scenario in which the vehicle is stationary (top-left sample) during observation phase. In this example, our method correctly deduces that the vehicle won't move forward by generating high confidence samples while considering the possibility that the vehicle might move forward as captured by two modes of prediction. When the true intention of the vehicle is unclear (top-right sample), our model produces samples with high confidence for both possibilities ahead of the vehicle.

E. Ablation Analysis

For ablation studies we report the results in terms of the displacement errors at some intermediate time steps (minDE@#s) and MR of the best of the $K = 6$ trajectories based on FDE.

a) Contribution of Modules: We begin our ablation study by measuring the contribution of individual modules on the overall performance. The results are shown in Table III. Our base model is a simple *trajectory* predictor that only uses the fine predictor module and we gradually add *goal* predictor, *waypoints* (coarse trajectory predictor), and *TTA* module. As expected, adding goals provides a constraint on the end point of the prediction, hence improving on both distance metrics and MR. Using additional waypoints further improves the performance, in particular on MR. The application of TTA at the end has a significant impact, especially in terms of minimizing the error propagation towards the end point of the predicted trajectories, as a result, it further lowers the error on most metrics. Overall, the improvements become more prominent, up to 6% on distance metrics and up to 10% on MR, with the increase of prediction horizon implying that simpler models are competitive for short futures but reliable long term prediction needs to utilize the benefits of the added modules.

b) Dynamic Goal Queries: In this section, we highlight the advantage of the proposed dynamic goal query module. We implement an alternative goal predictor, which

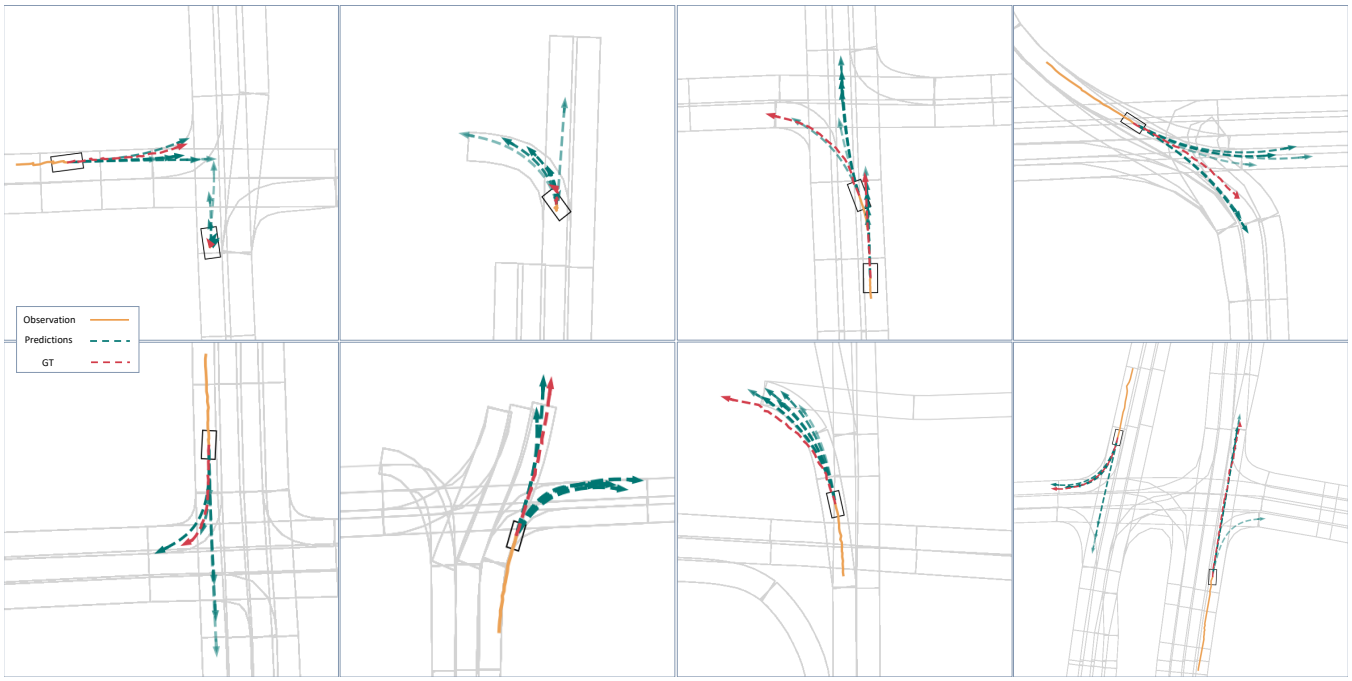

Fig. 3: Qualitative examples of our proposed method, DESTINE, on the Argoverse validation set. The observation is shown in orange, predictions are in shades of teal with opacity as confidence, and ground-truth is in red.

TABLE IV: Comparison of alternative goal predictor architectures.

Goal Decoder	layers	minDE@1s	minDE@2s	minDE@3s	MR
Learned Goal Predictor	2	0.53	0.79	0.95	0.088
Learned Goal Predictor	4	0.53	0.79	0.94	0.088
Learned Goal Predictor	6	0.53	0.79	0.94	0.087
Dynamic Goal Predictor	2	**0.52**	0.78	0.93	0.083
Dynamic Goal Predictor	4	**0.52**	**0.77**	**0.90**	**0.081**

TABLE V: Evaluation of different masking operations in TTA module.

TTA	minDE@1s	minDE@2s	minDE@3s	MR
-	0.52	0.79	0.93	0.084
1 layer/no-mask	0.53	0.81	0.97	0.094
1 layer/masked w history	0.52	0.79	0.92	0.086
1 layer/masked	**0.52**	**0.77**	**0.90**	**0.081**
2 layer/masked	0.52	0.78	0.91	0.081

we term *Learned Goal Predictor* that uses non-dynamic filter parameters learned during training time, similar to [20], [63]. In particular, we use 3 variants of *Learned Goal Predictor*, using an MLP network with 2, 4, and 6 layers. We compare these versions with our method *Dynamic Goal Predictor* with 2 and 4 transformer layers. Both *Learned Goal Predictor* and *Dynamic Goal Predictor* use the same input and they have similar structures for their outputs. Table IV summarizes the results of this study and shows the significant improvements of *Dynamic Queries* over *Learned Goal* decoders. We observe that the improvements on predictions are more significant towards the longer horizons, up to 7% on MR, pointing to the importance of goal prediction for minimizing long-term error propagation. Unlike the learned goal methods, dynamic queries performance boosts further by adding extra processing layer showing the capacity of the method for learning better goal predictions. Overall, we see that dynamic goal queries is a better solution for reliable prediction in longer term horizons.

c) Temporal Transductive Alignment (TTA) : To further probe into the contribution of TTA, we investigate different variants of TTA with *no-mask*, *mask with history* in which observation is also included in consistency refinement similar to [34], and our proposed approach *masked* with 1 and 2 layers of processing. As shown in Table V, without masking information, the use of TTA can have a negative effect and degrades the performance. This can be due to the fact that the additional transformer operation adds noise to the prediction as the model tries to establish connections across all time steps. Adding observation to the masking operation, no noticeable improvement is achieved. However, focusing only on predicted trajectories the performance is boosted across all time steps, in particular at the longest prediction horizon of 3s. This behavior is expected since TTA minimizes error propagation across time resulting in less deviation from the ground truth trajectory. Adding extra layer of processing, no improvements were achieved.

V. Conclusion

In this paper, we proposed DESTINE, a novel trajectory prediction model comprised of dynamic goal prediction, coarse-to-fine prediction, and temporal transductive alignment modules. The proposed goal prediction module uses dynamic queries to adaptively predict goals at inference time. The coarse-to-fine predictor provides intermediate waypoints serving as a condition for generating more stable trajectories. Lastly, the temporal alignment module refines the predicted trajectories minimizing error propagation in long-term prediction. Via empirical evaluations on different sets of the Argoverse benchmark dataset, we showed that our model achieves state-of-the-art performance on various metrics. We further highlighted the contributions of different proposed modules on the overall performance using ablation studies.

REFERENCES

[1] E. Amirloo, A. Rasouli, P. Lakner, M. Rohani, and J. Luo, "LatentFormer: Multi-agent transformer-based interaction modeling and trajectory prediction," *arXiv:2203.01880*, 2022.

[2] M. Arar, A. Shamir, and A. H. Bermano, "Learned queries for efficient local attention," in *CVPR*, 2022.

[3] I. Bae, J.-H. Park, and H.-G. Jeon, "Learning pedestrian group representations for multi-modal trajectory prediction," in *ECCV*, 2022.

[4] M. Bahari, S. Saadatnejad, A. Rahimi, M. Shaverdikondori, A. H. Shahidzadeh, S.-M. Moosavi-Dezfooli, and A. Alahi, "Vehicle trajectory prediction works, but not everywhere," in *CVPR*, 2022.

[5] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *ECCV*, 2020.

[6] S. Casas, C. Gulino, S. Suo, K. Luo, R. Liao, and R. Urtasun, "Implicit latent variable model for scene-consistent motion forecasting," in *ECCV*, 2020.

[7] Y. Chai, B. Sapp, M. Bansal, and D. Anguelov, "MultiPath: Multiple probabilistic anchor trajectory hypotheses for behavior prediction," *CoRL*, 2019.

[8] 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 *CVPR*, 2019.

[9] D. Choi and K. Min, "Hierarchical latent structure for multi-modal vehicle trajectory forecasting," in *ECCV*, 2022.

[10] A. Cui, S. Casas, A. Sadat, R. Liao, and R. Urtasun, "LookOut: Diverse multi-future prediction and planning for self-driving," in *ICCV*, 2021.

[11] H. Cui, V. Radosavljevic, F.-C. Chou, T.-H. Lin, T. Nguyen, T.-K. Huang, J. Schneider, and N. Djuric, "Multimodal trajectory predictions for autonomous driving using deep convolutional networks," in *ICRA*, 2019.

[12] F. Da and Y. Zhang, "Path-aware graph attention for HD maps in motion prediction," in *ICRA*, 2022.

[13] X. Dai, Y. Chen, J. Yang, P. Zhang, L. Yuan, and L. Zhang, "Dynamic detr: End-to-end object detection with dynamic attention," in *CVPR*, 2021.

[14] B. Dong, F. Zeng, T. Wang, X. Zhang, and Y. Wei, "SOLQ: Segmenting objects by learning queries," in *NeurIPS*, 2021.

[15] J. Du, J. Jiang, J. Zheng, H. Zhang, D. Huang, and Y. Lu, "Improving computation and memory efficiency for real-world transformer inference on gpus," *ACM Transactions on Architecture and Code Optimization*, vol. 20, no. 4, pp. 1–22, 2023.

[16] L. Fang, Q. Jiang, J. Shi, and B. Zhou, "TPNET: Trajectory proposal network for motion prediction," in *CVPR*, 2020.

[17] Y. Fang, S. Yang, X. Wang, Y. Li, C. Fang, Y. Shan, B. Feng, and W. Liu, "Instances as queries," in *ICCV*, 2021.

[18] J. Gao, C. Sun, H. Zhao, Y. Shen, D. Anguelov, C. Li, and C. Schmid, "VectorNet: Encoding hd maps and agent dynamics from vectorized representation," in *CVPR*, 2020.

[19] R. Girgis, F. Golemo, F. Codevilla, M. Weiss, J. A. D'Souza, S. E. Kahou, F. Heide, and C. Pal, "Latent variable sequential set transformers for joint multi-agent motion prediction," in *ICLR*, 2022.

[20] J. Gu, C. Sun, and H. Zhao, "DenseTNT: End-to-end trajectory prediction from dense goal sets," in *ICCV*, 2021.

[21] Y. Han, G. Huang, S. Song, L. Yang, H. Wang, and Y. Wang, "Dynamic neural networks: A survey," *PAMI*, vol. 44, no. 11, pp. 7436–7456, 2021.

[22] D. Helbing and P. Molnar, "Social force model for pedestrian dynamics," *Physical Review E*, vol. 51, no. 5, pp. 4282–4286, 1995.

[23] B. Hu and T.-J. Cham, "Entry-flipped transformer for inference and prediction of participant behavior," in *ECCV*, 2022.

[24] Z. Huang, X. Mo, and C. Lv, "Multi-modal motion prediction with transformer-based neural network for autonomous driving," in *ICRA*, 2022.

[25] X. Jia, B. De Brabandere, T. Tuytelaars, and L. V. Gool, "Dynamic filter networks," in *NeurIPS*, 2016.

[26] R. Karim, H. Zhao, R. P. Wildes, and M. Siam, "MED-VT: Multiscale encoder-decoder video transformer with application to object segmentation," in *CVPR*, 2023.

[27] B. Kim, S. H. Park, S. Lee, E. Khoshimjonov, D. Kum, J. Kim, J. S. Kim, and J. W. Choi, "LaPred: Lane-aware prediction of multi-modal future trajectories of dynamic agents," in *CVPR*, 2021.

[28] N. Kitaev, L. Kaiser, and A. Levskaya, "Reformer: The efficient transformer," in *ICLR*, 2019.

[29] V. Kosaraju, A. Sadeghian, R. Martin-Martin, I. Reid, H. Rezatofighi, and S. Savarese, "Social-BiGAT: Multimodal trajectory forecasting using Bicycle-GAN and graph attention networks," in *NeurIPS*, 2019.

[30] P. Kothari, B. Sifringer, and A. Alahi, "Interpretable social anchors for human trajectory forecasting in crowds," in *CVPR*, 2021.

[31] M. Lee, S. S. Sohn, S. Moon, S. Yoon, M. Kapadia, and V. Pavlovic, "MUSE-VAE: Multi-scale VAE for environment-aware long term trajectory prediction," in *CVPR*, 2022.

[32] N. Lee, W. Choi, P. Vernaza, C. B. Choy, P. H. S. Torr, and M. Chandraker, "DESIRE: Distant future prediction in dynamic scenes with interacting agents," in *CVPR*, 2017.

[33] L. Li, M. Pagnucco, and Y. Song, "Graph-based spatial transformer with memory replay for multi-future pedestrian trajectory prediction," in *CVPR*, 2022.

[34] S. Li, Y. Zhou, J. Yi, and J. Gall, "Spatial-temporal consistency network for low-latency trajectory forecasting," in *ICCV*, 2021.

[35] M. Liang, B. Yang, R. Hu, Y. Chen, R. Liao, S. Feng, and R. Urtasun, "Learning lane graph representations for motion forecasting," in *ECCV*, 2020.

[36] Y. Liu, J. Zhang, L. Fang, Q. Jiang, and B. Zhou, "Multimodal motion prediction with stacked transformers," in *CVPR*, 2021.

[37] I. Loshchilov and F. Hutter, "SGDR: Stochastic gradient descent with warm restarts," in *ICLR*, 2017.

[38] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *ICLR*, 2019.

[39] K. Mangalam, Y. An, H. Girase, and J. Malik, "From goals, waypoints & paths to long term human trajectory forecasting," in *ICCV*, 2021.

[40] A. Mohamed, K. Qian, M. Elhoseiny, and C. Claudel, "Social-STGCNN: A social spatio-temporal graph convolutional neural network for human trajectory prediction," in *CVPR*, 2020.

[41] J. Ngiam, V. Vasudevan, B. Caine, Z. Zhang, H.-T. L. Chiang, J. Ling, R. Roelofs, A. Bewley, C. Liu, A. Venugopal, *et al.*, "Scene transformer: A unified architecture for predicting future trajectories of multiple agents," in *ICLR*, 2022.

[42] S. H. Park, G. Lee, J. Seo, M. Bhat, M. Kang, J. Francis, A. Jadhav, P. P. Liang, and L.-P. Morency, "Diverse and admissible trajectory forecasting through multimodal context understanding," in *ECCV*, 2020.

[43] A. Rasouli and I. Kotseruba, "PedFormer: Pedestrian behavior prediction via cross-modal attention modulation and gated multitask learning," in *ICRA*, 2023.

[44] A. Rasouli, M. Rohani, and J. Luo, "Bifold and semantic reasoning for pedestrian behavior prediction," in *ICCV*, 2021.

[45] X. Ren, T. Yang, L. E. Li, A. Alahi, and Q. Chen, "Safety-aware motion prediction with unseen vehicles for autonomous driving," in *ICCV*, 2021.

[46] A. Rudenko, L. Palmieri, and K. O. Arras, "Joint long-term prediction of human motion using a planning-based social force approach," in *ICRA*, 2018.

[47] T. Salzmann, B. Ivanovic, P. Chakravarty, and M. Pavone, "Trajectron++: Multi-agent generative trajectory forecasting with heterogeneous data for control," in *ECCV*, 2020.

[48] F. Schmidt, "Generalization in generation: A closer look at exposure bias," *arXiv:1910.00292*, 2019.

[49] L. Shi, L. Wang, C. Long, S. Zhou, M. Zhou, Z. Niu, and G. Hua, "SGCN: Sparse graph convolution network for pedestrian trajectory prediction," in *CVPR*, 2021.

[50] B. Varadarajan, A. Hefny, A. Srivastava, K. S. Refaat, N. Nayakanti, A. Cornman, K. Chen, B. Douillard, C. P. Lam, D. Anguelov, and B. Sapp, "MultiPath++: Efficient information fusion and trajectory aggregation for behavior prediction," in *ICRA*, 2022.

[51] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *NeurIPS*, 2017.

[52] C. Wang, Y. Wang, M. Xu, and D. J. Crandall, "Stepwise goal-driven networks for trajectory prediction," *RAL*, vol. 7, no. 2, pp. 2716–2723, 2022.

[53] J. Wang, T. Ye, Z. Gu, and J. Chen, "LTP: Lane-based trajectory prediction for autonomous driving," in *CVPR*, 2022.

[54] C. Xu, M. Li, Z. Ni, Y. Zhang, and S. Chen, "GroupNet: Multiscale hypergraph neural networks for trajectory prediction with relational reasoning," in *CVPR*, 2022.

[55] C. Xu, W. Mao, W. Zhang, and S. Chen, "Remember intentions: Retrospective-memory-based trajectory prediction," in *CVPR*, 2022.

[56] K. Yamaguchi, A. C. Berg, L. E. Ortiz, and T. L. Berg, "Who are you with and where are you going?" in *CVPR*, 2011.
[57] B. Yang, G. Bender, Q. V. Le, and J. Ngiam, "Condconv: Conditionally parameterized convolutions for efficient inference," *NeurIPS*, 2019.
[58] Y. Yao, E. Atkins, M. Johnson-Roberson, R. Vasudevan, and X. Du, "BiTraP: Bi-directional pedestrian trajectory prediction with multi-modal goal estimation," *RAL*, vol. 6, no. 2, pp. 1463–1470, 2021.
[59] C. Yu, X. Ma, J. Ren, H. Zhao, and S. Yi, "Spatio-temporal graph transformer networks for pedestrian trajectory prediction," in *ECCV*, 2020.
[60] R. Yu and Z. Zhou, "Towards robust human trajectory prediction in raw videos," in *IROS*, 2021.
[61] Y. Yuan, X. Weng, Y. Ou, and K. M. Kitani, "AgentFormer: Agent-aware transformers for socio-temporal multi-agent forecasting," in *ICCV*, 2021.
[62] L. Zhang, P.-H. Su, J. Hoang, G. C. Haynes, and M. Marchetti-Bowick, "Map-adaptive goal-based trajectory prediction," in *CoRL*, 2021.
[63] H. Zhao, J. Gao, T. Lan, C. Sun, B. Sapp, B. Varadarajan, Y. Shen, Y. Shen, Y. Chai, C. Schmid, *et al.*, "TNT: Target-driven trajectory prediction," in *CoRL*, 2020.
[64] F. Zheng, L. Wang, S. Zhou, W. Tang, Z. Niu, N. Zheng, and G. Hua, "Unlimited neighborhood interaction for heterogeneous trajectory prediction," in *ICCV*, 2021.
[65] J. Zhou, V. Jampani, Z. Pi, Q. Liu, and M.-H. Yang, "Decoupled dynamic filter networks," in *CVPR*, 2021.
[66] Z. Zhou, L. Ye, J. Wang, K. Wu, and K. Lu, "HiVT: Hierarchical vector transformer for multi-agent motion prediction," in *CVPR*, 2022.