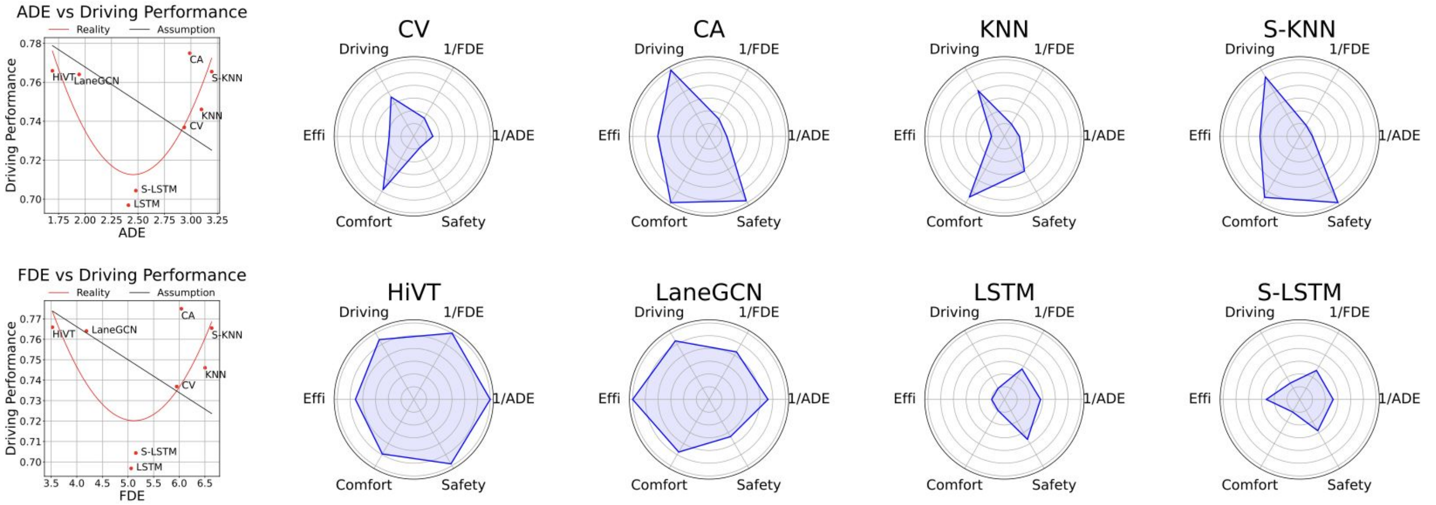
Prediction

Quickfire paper summaries

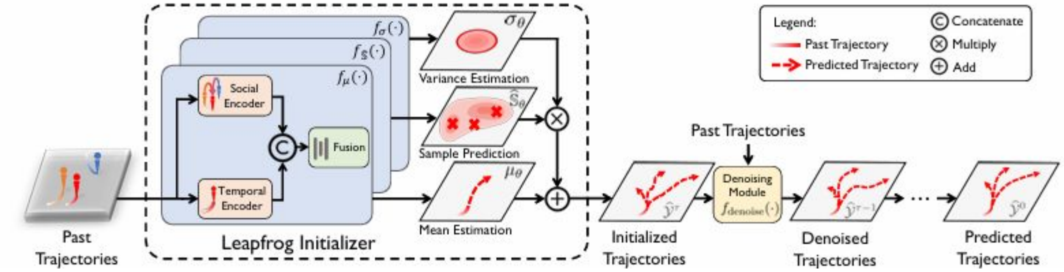
What Truly Matters in Trajectory Prediction for Autonomous Driving

Secret Sauce: Balancing computation speed vs prediction accuracy

Single sentence describing the work: The paper describes the importance of trajectory planning while comparing the prediction gap to real time changes being made.



Leapfrog Diffusion Model for Stochastic Trajectory Prediction



- Focuses on stochasticity for trajectory prediction
- Idea is to learn a rough, yet sufficiently expressive distribution to initialize denoised future trajectories instead of plain gaussian
- Tackles two problems:
  - To increase the real time inference
  - Capture the sufficient multi-modality

Leapfrog Diffusion Model for Stochastic Trajectory Prediction (Isaac Remy)

- **Problem:** diffusion models show great promise for human trajectory prediction, but denoising inference is too slow for real-time deployment
- **Approach:** Train an “initializer“ that predicts the mean, variance, and future position of agents from noisy initalization
- **Claim:** Leapfrog significantly shortens inference time online
- **Evaluation:** They look at prediction accuracy and inference time on multiple human-trajectory datasets, showing their approach outperforms SOTA methods.

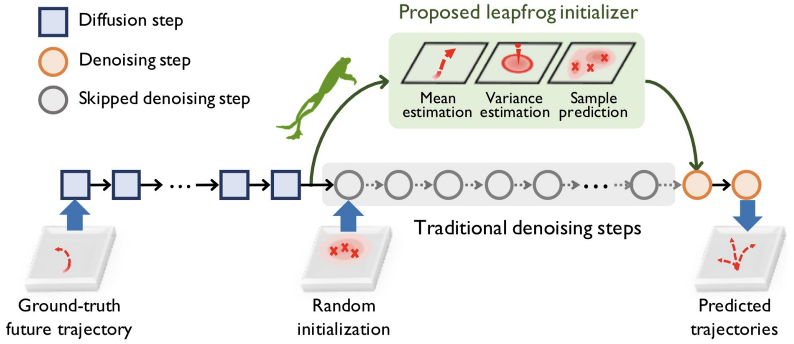
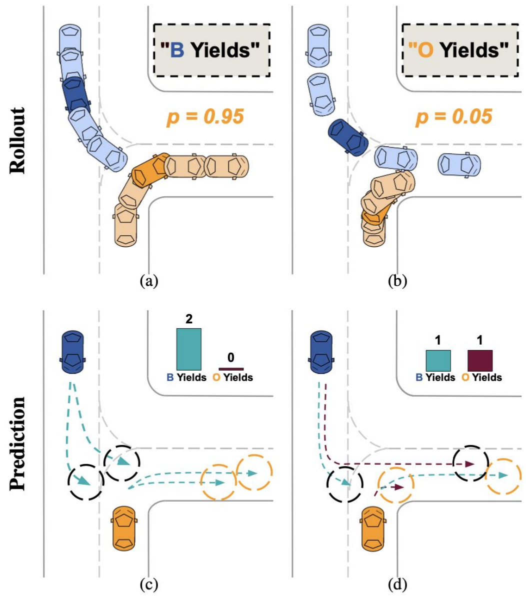


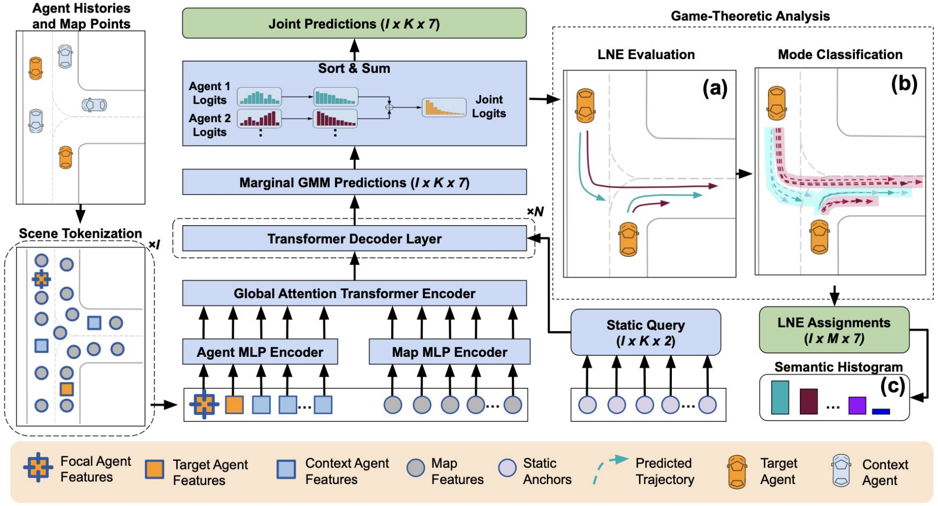
Figure 1. Leapfrog diffusion model uses the leapfrog initializer to estimate the denoised distribution and substitute a long sequence of traditional denoising steps, accelerating inference and maintaining representation capacity.

NashFormer: Leveraging Local Nash Equilibria for Semantically Diverse Trajectory Prediction (Jake Gonzales)

- Present NashFormer: a framework for trajectory prediction that leverages game-theoretic IRL to improve coverage of multi-modal predictions



Problem



Solution and Architecture

Robots That Can See: Leveraging Human Pose for Trajectory Prediction (Nivii Kalavakonda)

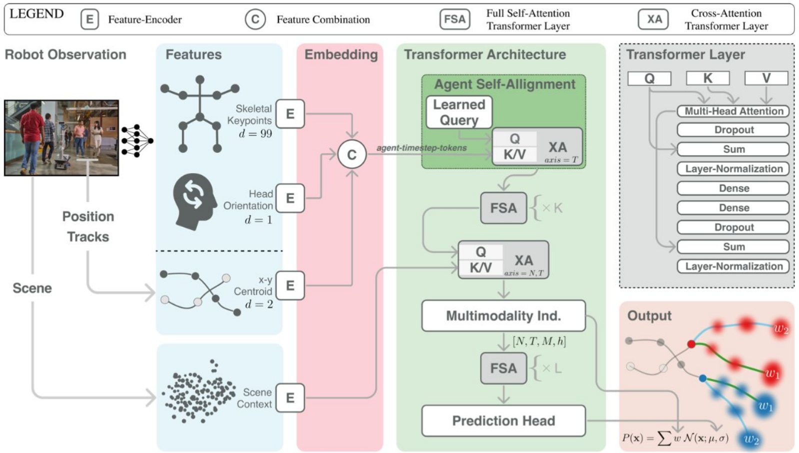
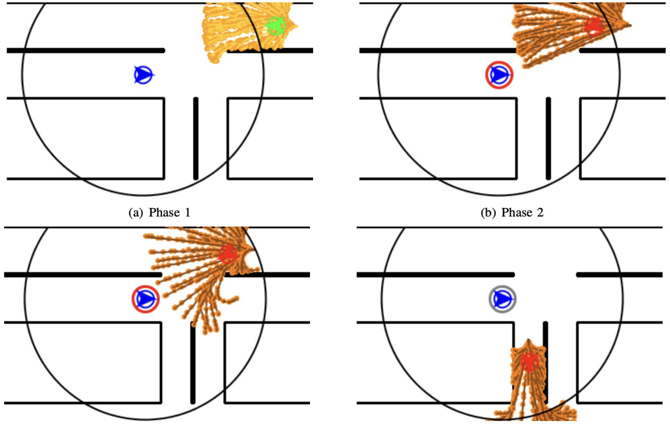


Fig. 3: **Overview of the HST architecture.** From the robot's sensors we extract the scene context, the historic position tracks of each agent, and vision based skeletal keypoints/head orientation when feasible. All features are encoded individually before the agent features are combined via cross-attention (XA) using a learned query tensor. The resulting agent-timestep-tokens is passed to our Agent Self-Alignment layer which enables the use of subsequent full self-attention (FSA) layers. Embedded scene context is attended to via cross-attention (XA). After multimodality is induced and further FSA layers the model outputs the parameters of a Normal distribution for each agent at each prediction timestep. We can represent the full output structure as a Gaussian Mixture Model (formula in bottom right) over all possible futures where the mixture coefficients w come from the Multimodality Induction. Both cross-attention (XA) and full self-attention layers use the Transformer layer (top right) with different input configurations for Query (Q), Key (K), and Value (V).

- **Goal**: Predict future human trajectories from input features including position history for humans, head orientations (when available) and 3D skeletal keypoints using a Human Scene Transformer
- **Result**: Found new agents with limited historical data as a major contributor to error and demonstrate the complementary nature of 3D skeletal poses in reducing prediction error in such challenging scenarios

Threat assessment design for driver assistance system at intersections

- Model using support vector machine (intention classification) and rapidly exploring trees for threat assessment and detection module in a vehicle
- Reachability analysis to predict collision
- If collision is imminent, determine evasive maneuver to minimize ‘threat’ of collision

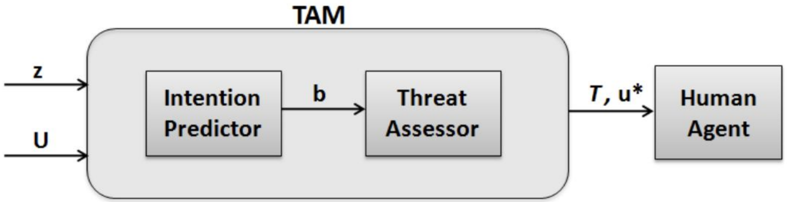


(a) Phase 1

(b) Phase 2

(c) Phase 3

(d) Phase 4



What Truly Matters in Trajectory Prediction for Autonomous Driving?

Phong Tran, Haoran Wu, Cunjun Yu, Panpan Cai, Sifa Zheng, David Hsu

- Standard behavior prediction metrics evaluate predictions against recorded future behavior
- When deployed as part of the autonomous system, the predictions will be used to influence the robot's plan, which will then influence other agents
- It is important to be aware of this interaction when evaluating prediction systems

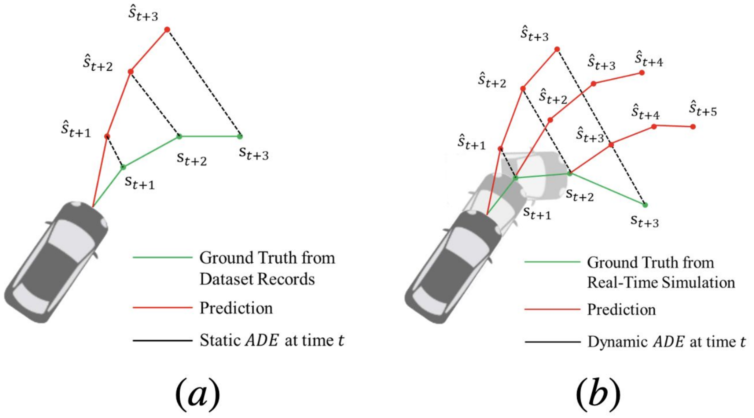


Figure 2: Dynamics Gap. (*a*) In static evaluation, the agent's motion is determined and unaffected by predictors. (*b*) In the real-world, different predictors result in varied behaviors of the agent, which directly affects the ground truth of prediction.

Ethan Pronovost