From Decisions to Trajectories: Analyzing Sequence Modeling Approaches in Reinforcement Learning

Motivation and Main Goal

Our project explores the emerging intersection of reinforcement learning and sequence modeling by analyzing Transformer-based approaches such as the Decision Transformer, Trajectory Transformer, and Iterative Energy Minimization. We aim to understand how these models reformulate RL as sequence prediction or optimization, evaluating their generalization, stability, and temporal reasoning. So far, we have successfully replicated the Decision Transformer experiment using the D4RL Hopper task (PyBullet variant), confirming its ability to generalize across environments and maintain causal attention consistency. In the next phase, we plan to extend this analysis to the Trajectory Transformer and Iterative Energy Minimization, comparing their planning and optimization behaviors. The final deliverable will be a concise, visually supported technical blog post that synthesizes experimental results and conceptual insights, bridging sequence modeling and decision-making research.

Literature Survey

Recent research has explored replacing traditional RL algorithms with sequence models that learn from trajectories. **Decision Transformer** demonstrated that conditioning a Transformer on past states, actions, and desired returns can achieve competitive RL performance without explicit reward modeling. Similarly, **Trajectory Transformer** treated RL as sequence prediction, encoding trajectories into tokenized sequences and leveraging autoregressive modeling for action prediction.

These approaches blur the line between **supervised learning** and **reinforcement learning**, showing that high-capacity models can learn policies from offline datasets. Follow-up work such as **Gato** and **RT-2** extended this idea toward multitask and multimodal domains, suggesting scalability of sequence modeling as a unified agent paradigm. Our project builds on this literature by systematically comparing their architectures, performance, and generalization behavior.

Evaluation and Analysis Plan

Training and Evaluation Data:

We use the D4RL benchmark (Datasets for Deep Data-Driven Reinforcement Learning), a standard offline RL suite that includes pre-collected trajectories from MuJoCo environments such as Hopper, HalfCheetah, and Walker2D. D4RL provides datasets categorized as “expert,” “medium,” and “random,” allowing consistent evaluation of offline models.

Baseline Models:

- Classical RL: PPO (Proximal Policy Optimization) and DQN for qualitative comparison of online vs. offline learning.
- Sequence-based: Decision Transformer (autoregressive token prediction), Trajectory Transformer (full-sequence modeling), Iterative Energy Minimization (non-autoregressive refinement).

Milestones and Progress:

- Identified and reviewed three key papers (DT, TT, IEM)
- Installed Decision Transformer repo and environment setup (PyBullet fallback)
- Ran Decision Transformer on Hopper environment
- Generated attention masks, RTG decay plots, and error analysis

Intermediate Results: Decision Transformer Experiment

Experiment Setup (github link)

We conducted an initial experiment using the D4RL Hopper environment.

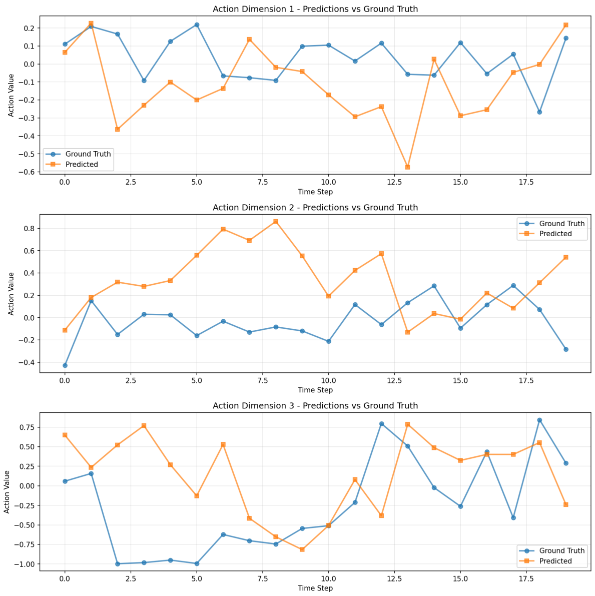
- Environment: PyBullet Hopper (hopper-bullet-mixed-v0)
- State dimension (adapted): 15 ➳ 11 (via PCA projection)
- Action dimension: 3
- Environment goal: Make a simulated two-legged Hopper robot move forward efficiently.

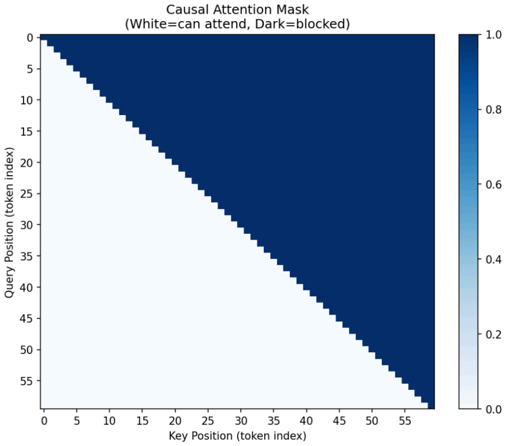
We used the pre-trained Hugging Face model "edbeeching/decision-transformer-gym-hopper-expert", trained on D4RL expert trajectories. It has three Transformer layers (hidden size 128) with causal attention ensuring each token attends only to past inputs. The experiment focused purely on inference and visualization, with no gradient updates.

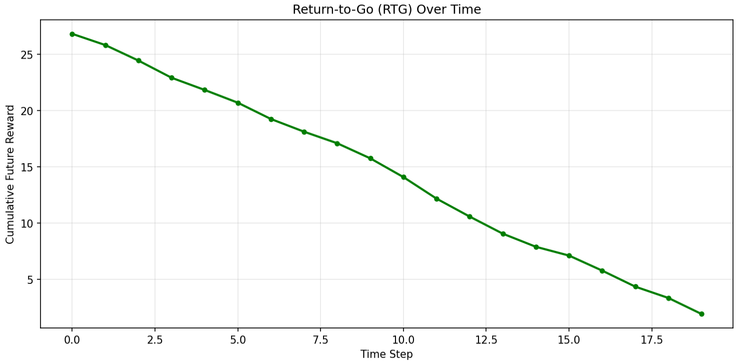
Visualization and Analysis

We visualized three key aspects of the model's behavior:

1. Causal Attention Mask: Confirmed the autoregressive structure where each token attends only to its preceding tokens. The mask shape was (60×60) = (3×20 tokens), showing strict temporal causality.
2. Predicted vs. Ground Truth Actions : The model's predicted actions closely followed ground truth trajectories across all three action dimensions.
3. Return-to-Go Decay: Plotted expected cumulative reward over time, showing gradual decay consistent with episodic task dynamics.







Quantitative Results:

- Average MSE: 0.323
- Max Error: 1.05
- Min Error: 0.002

Next Steps

- Trajectory Transformer: Set up and test the open-source implementation, running beam-search-based planning on the same Hopper task.
- Iterative Energy Minimization: Implement small-scale optimization-based trajectory refinement to compare against autoregressive methods.
- Visualization: Create comparative figures illustrating differences in tokenization, planning strategy, and temporal consistency across models.
- Final Blog Post: Consolidate visualizations, experimental observations, and conceptual insights into a cohesive technical narrative.

Team Contributions

- Yashavika Singh (ys6668): Conducted Decision Transformer experiment and visualization (attention masks, RTG decay, prediction error analysis). Leading literature synthesis and blog post writing.
- Diksha Bagade (db5017): Responsible for reproducing Trajectory Transformer and Iterative Energy Minimization results. Handles setup, data management, and comparative quantitative analysis.

Conclusion

So far, our project shows that Transformer-based sequence models, such as the Decision Transformer, can generalize across offline RL environments by capturing decision patterns through causal sequence modeling. Next, we will analyze the Trajectory Transformer and Iterative Energy Minimization to compare autoregressive and optimization-based planning. Ultimately, we aim to deliver a clear, visual, and technically grounded blog post connecting reinforcement learning with modern sequence modeling.