Transformers, take the Wheel! Sequence Modeling in Offline RL

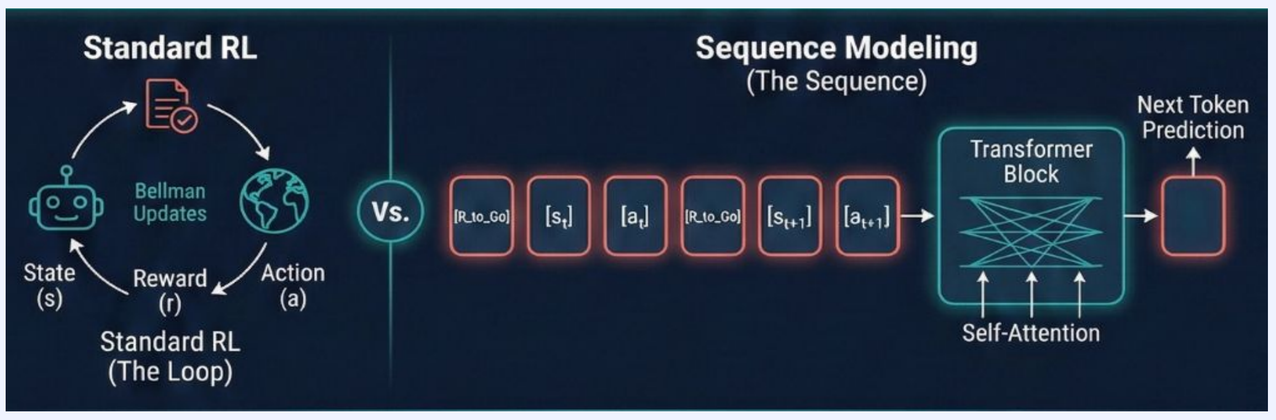
DSGA 1011 Fundamentals of Natural Language Processing

Team: Diksha Bagade, Yashavika Singh

Abstract

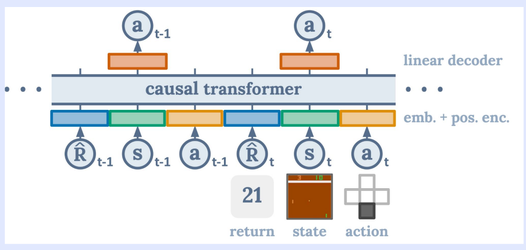
Reinforcement Learning (RL) has traditionally relied on value estimation and Bellman updates, which are often unstable and difficult to tune.
This project explores a paradigm shift: treating RL as a Sequence Modeling problem. We analyze and replicate three Transformer-based approaches—Decision Transformer (DT), Trajectory Transformer (TT), and Iterative Energy Minimization (IEM)—to understand how language modeling architectures can solve decision-making tasks.

The Paradigm Shift

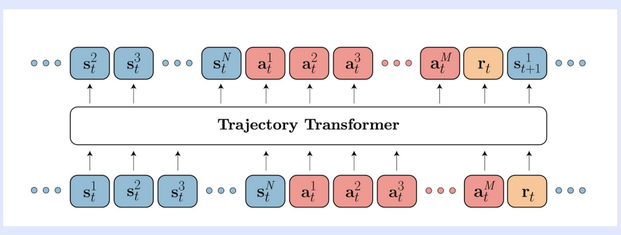


Papers Selected

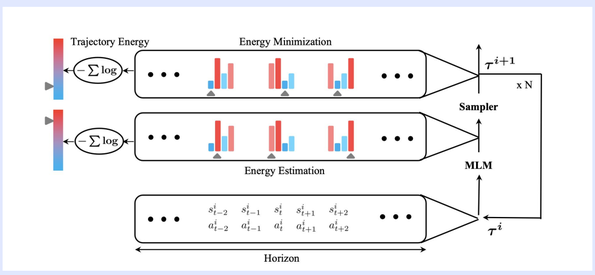
Decision Transformer (DT): It establishes the baseline proof-of-concept models Reinforcement Learning as a Sequential modeling task.
Architecture used: causal GPT



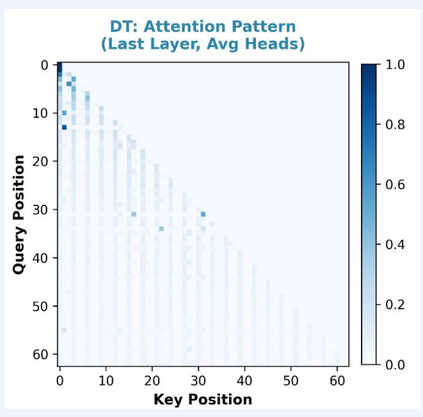
Trajectory Transformer (TT): IT accepts the premise of DT (RL is Sequence Modeling) but critiques the "blind" generation. To actively plan into the future, it adapts the NLP concept of Beam Search.
Architecture used: casual GPT

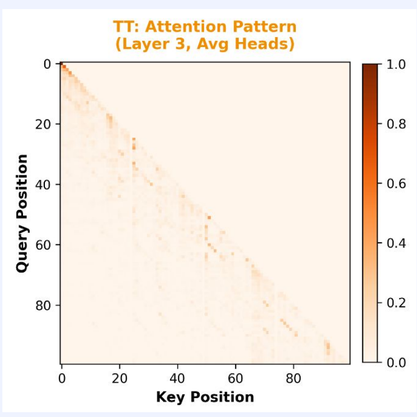


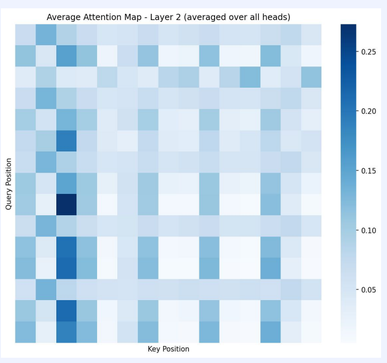
Iterative Energy Minimization (IEM): "The Refiner" – Uses a BERT-like masked model to iteratively "denoise" and optimize a full plan at once, minimizing a learned energy function.
Architecture used: BERT



Inside the Black Box: Attention Analysis







DT: Vertical attention stripes confirm the model explicitly "checks" the desired future reward before committing to an action.
TT: Strong diagonal banding reveals it focuses on immediate past context over long term past.
IEM: Distributed grid-like attention states each position attends broadly across past AND future

Results and Benchmarks

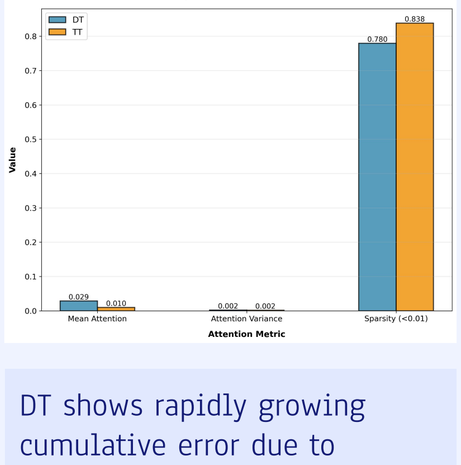
Method	Type	Locomotion	Atari	AntMaze	Sparse Rewards
BC	Imitation	47.7	60.9	10.9	Poor
CQL	TD Learning	77.6	107.2	44.9	Fails
IQL	TD Learning	~75	-	**63.2**	Moderate
DT	Seq. Modeling	74.7	97.9	11.8	Robust
TT	Seq. Modeling	**78.9**	-	-	Good
LEAP	Energy-Based	-	**132.6**	-	Good

[table] Sequence models match TD learning on dense rewards, dominate on sparse rewards where TD fails, and hybrid LEAP achieves best overall performance.

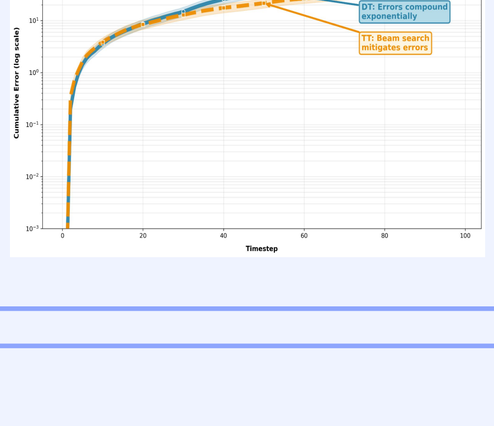
Transformers fundamentally solve the credit assignment and sparse reward problems of TD , while hybrid and energy-based extensions unlock capabilities like trajectory stitching and task composition that pure methods cannot achieve.

Performance analysis

DT and TT share autoregressive architectures that commit to tokens sequentially, making their attention patterns and error accumulation directly comparable Both are evaluated on HalfCheetah-v4 for fair comparison.



DT has higher mean attention and is less sparse, indicating it allocates attention more broadly across tokens rather than focusing on key elements. This more distributed attention in DT may contribute to its higher error accumulation.



DT shows rapidly growing cumulative error due to autoregressive compounding, while TT shows lower, more stable error accumulation because beam search mitigates compounding by exploring multiple trajectory hypotheses.

Limitations and Future Works

Sequence models struggle with trajectory stitching and require **significantly more compute** than feedforward policies. Future work should explore lightweight architectures for real-time control and principled methods for combining sequence planning with dynamic programming.

Conclusion

The convergence of NLP and RL provides a unified framework where trajectories are treated as sentences, offering stability that traditional dynamic programming lacks. The future lies in hybrid architectures: combining sequence models' distributional robustness with Q-learning's trajectory stitching, and LEAP's iterative refinement for composable, adaptable planning.

References

Chen, L., et al. (2021). Decision Transformer: Reinforcement Learning via Sequence Modeling. NeurIPS.
Janner, M., et al. (2021). Offline Reinforcement Learning as One Big Sequence Modeling Problem. NeurIPS.
Chen, H., et al. (2023). Planning with Sequence Models through Iterative Energy Minimization. ICLR.