Decision Transformer: Reinforcement Learning via Sequence modeling

Ben Kabongo

January 27, 2024

Master MVA, ENS Paris-Saclay

Table of contents

Introduction

Overview

Decision Transformer:

- is an architecture that transforms the reinforcement learning problem into conditional sequence modeling
- simply produces optimal actions by relying on a causally masked Transformer
- meets or exceeds the performance of state-of-the-art model-free offline reinforcement learning databases [12]

Problem definition

Markov Decision Process and Reinforcement Learning

A MDP is a tuple $\big(\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}\big)$:

- $\mathcal{S}$: set of states.
- $\mathcal{A}$: set of actions.
- $\mathcal{P}$: transition function $\mathcal{T} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ $\mathcal{T}(s, a, s') = \mathcal{P}(s'|s, a)$
- $\mathcal{R}$: reward function $\mathcal{R} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$

Goal: maximize $\mathbb{E}\left[\sum_t \gamma^t r_t\right]$ [63]

Policy π: deterministic $\pi : \mathcal{S} \to \mathcal{A}$ or probabilistic $\pi : \mathcal{S} \times \mathcal{A} \to [0, 1]$

Trajectory τ: $\tau = \big(s_0, a_0, r_0, s_1, a_1, r_1, \cdots, s_T, a_T, r_T\big)$

Online RL: learning with arbitrary policy trajectory data.

Transformer

- Queries, Keys, Values: $Q = XW^Q$, $K = XW^K$, $V = XW^V$
- $\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$
- $\text{MultiHead}(Q, K, V) = \text{Concat}(head_1, ..., head_h)W^O$
  - $head_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$
- **Encoder**: learn richer representations
- **Decoder**: perform better on generative tasks like next token prediction

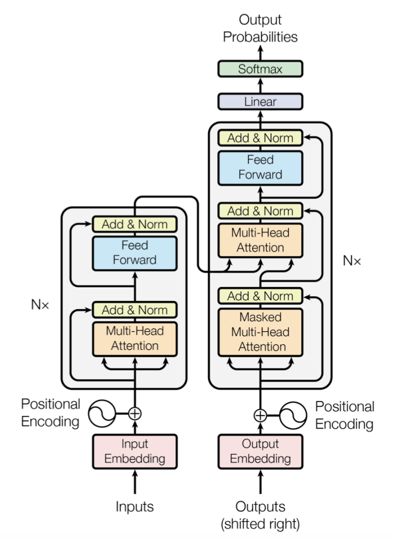

Figure 1: Transformer [64]

Related work

Related work

Attention and transformer models

- Transformer [64] in NLP [16] [53] in Computer Vision [11] [17]
- Transformer in RL with actor-critical algorithms [71] [57] [49]
- Transformer in RL instead of RNN [14] [1]

Offline reinforcement learning

- Offline learning is sensitive to distribution change : [40] [23] [38] [59] [36] [70]
- Other work explores learning a large distribution of behaviors from an offline dataset
- Likelihood-based approaches [3] [10] [52] [60]
- Mutual information approaches [19] [44] [58]

Decision Transformer [12]

Method

Trajectory representation: Let $\hat{R}_t = \sum_{t=1}^{T} r_t$

$$\tau = (\hat{R}_1, s_1, a_1, \hat{R}_2, s_2, a_2, \cdots, \hat{R}_T, s_T, a_T)$$

Architecture:

- **Input**: K last time steps = 3K tokens
- **Modalities**: return-to-go, state and action
- Embedding per time and embedding for modalities
- The tokens are then processed by a GPT model [53]

Training:

- Predicts future action tokens through autoregressive modeling
- **Loss**: cross-entropy (discrete), MSE (continuous)

Architecture

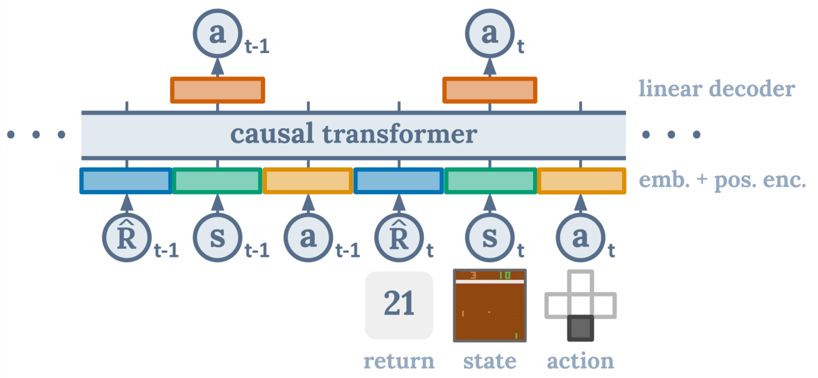

Figure 2: Decision Transformer Architecture [12]

Algorithm

Algorithm 1 Decision Transformer Pseudocode (for continuous actions)

```
# R, s, a, t: returns-to-go, states, actions, or timesteps
# transformer: transformer with causal masking (GPT)
# embed_s, embed_a, embed_R: linear embedding layers
# embed_t: learned episode positional embedding
# pred_a: linear action prediction layer

# main model
def DecisionTransformer(R, s, a, t):
    # compute embeddings for tokens
    pos_embedding = embed_t(t)  # per-timestep (note: not per-token)
    s_embedding = embed_s(s) + pos_embedding
    a_embedding = embed_a(a) + pos_embedding
    R_embedding = embed_R(R) + pos_embedding

    # interleave tokens as (R_1, s_1, a_1, ..., R_K, s_K)
    input_embeds = stack(R_embedding, s_embedding, a_embedding)

    # use transformer to get hidden states
    hidden_states = transformer(input_embeds=input_embeds)

    # select hidden states for action prediction tokens
    a_hidden = unstack(hidden_states).actions

    # predict action
    return pred_a(a_hidden)

# training loop
for (R, s, a, t) in dataloader:  # dims: (batch_size, K, dim)
    a_preds = DecisionTransformer(R, s, a, t)
    loss = mean((a_preds - a)**2)  # L2 loss for continuous actions
    optimizer.zero_grad(); loss.backward(); optimizer.step()

# evaluation loop
target_return = 1  # for instance, expert-level return
R, s, a, t, done = [target_return], [env.reset()], [], [1], False
while not done:  # autoregressive generation/sampling
    # sample next action
    action = DecisionTransformer(R, s, a, t)[-1]  # for cts actions
    new_s, r, done, _ = env.step(action)

    # append new tokens to sequence
    R = R + [R[-1] - r]  # decrement returns-to-go with reward
    s, a, t = s + [new_s], a + [action], t + [len(R)]
    R, s, a, t = R[-K:], ...  # only keep context length of K
```

Figure 3: Decision Transformer Algorithm [12]

Evaluation

Environnements and datasets

- **MDPs**: $\mathcal{S}$ and $\mathcal{A}$ discretes, $\mathcal{P}$ and $\mathcal{R}$ deterministics
- **States**: 10^2, 10^3, 10^4, 10^5, 10^6
- **Actions**: 2, 3, 4, 5, 10, 20, 50, 100
- **Rewards**: 3, 4, 5, 10
- **Trajectory generations**
  - Learn an optimal policy
  - Trajectories generated by alternating optimal policy and random policy

Decision Transformer on different MDPs

	States			
Actions	10^2	10^3	10^4	10^5
2	69.55 ± 27.54	52.12 ± 14.23	48.77 ± 8.34	52.48 ± 5.15
3	72.89 ± 17.20	58.48 ± 10.97	50.23 ± 8.04	50.33 ± 0.96
4	74.65 ± 24.71	52.16 ± 2.25	50.88 ± 5.08	54.31 ± 5.54
5	65.82 ± 19.95	54.83 ± 8.61	58.76 ± 12.39	52.71 ± 2.30
10	97.60 ± 1.80	39.05 ± 16.93	51.82 ± 5.27	50.82 ± 0.81
20	84.61 ± 20.48	42.49 ± 5.30	52.89 ± 8.63	53.04 ± 4.25
50	71.10 ± 29.33	57.54 ± 13.29	52.36 ± 2.83	49.22 ± 1.06
100	-	-	-	59.35 ± 18.55

Table 1: Decision Transformer scores on MDPs for different configurations of number of states, number of actions. We report the mean and standard deviation for different numbers of rewards.

Decision Transformer structure

	Blocks			
h	**1**	**2**	**4**	**6**
1	51.53 ± 22.44	69.90 ± 19.49	56.49 ± 10.30	50.12 ± 19.63
2	53.02 ± 06.10	59.12 ± 16.29	71.40 ± 21.51	76.59 ± 19.99
4	49.07 ± 06.03	71.15 ± 20.57	53.06 ± 02.11	81.47 ± 17.57
8	50.24 ± 08.90	58.96 ± 16.33	67.00 ± 23.05	49.40 ± 05.16

Table 2: Decision Transformer scores on MDPs for different configurations of number of blocks, number of heads and embedding dimension.

Conclusion

Conclusion

- Decision Transformer: Reinforcement Learning via sequence modeling
- We decided to evaluate Decision Transformer on simple MDPs
- Experiments show that Decision Transformer can be an architecture of choice for tackling offline reinforcement learning problems
- It is important to ensure the reliability of this data for real applications
- Current and future work further exploits how to effectively integrate transformers into reinforcement learning

Thank you for your attention !

References

References i

J. Abramson, A. Ahuja, I. Barr, A. Brussee, F. Carnevale, M. Cassin, R. Chhaparia, S. Clark, B. Damoc, A. Dudzik, et al.
Imitating interactive intelligence.
arXiv preprint arXiv:2012.05672, 2020.

R. Agarwal, D. Schuurmans, and M. Norouzi.
An optimistic perspective on offline reinforcement learning.
In *International Conference on Machine Learning*, 2020.

A. Ajay, A. Kumar, P. Agrawal, S. Levine, and O. Nachum.
Opal: Offline primitive discovery for accelerating offline reinforcement learning.
arXiv preprint arXiv:2010.13611, 2020.

J. A. Arjona-Medina, M. Gillhofer, M. Widrich, T. Unterthiner, J. Brandstetter, and S. Hochreiter.
Rudder: Return decomposition for delayed rewards.
arXiv preprint arXiv:1806.07857, 2018.

J. L. Ba, J. R. Kiros, and G. E. Hinton.
Layer normalization.
arXiv preprint arXiv:1607.06450, 2016.

M. G. Bellemare, Y. Naddaf, J. Veness, and M. Bowling.
The arcade learning environment: An evaluation platform for general agents.
Journal of Artificial Intelligence Research, 47:253–279, 2013.

References iii

R. Bellman.
A markovian decision process.
Journal of Mathematics and Mechanics, 6(5):679–684, 1957.
JSTOR 24900506.

G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba.
Openai gym.
arXiv preprint arXiv:1606.01540, 2016.

T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, and e. a. Askell, Amanda.
Language models are few-shot learners.
arXiv preprint arXiv:2005.14165, 2020.

V. Campos, A. Trott, C. Xiong, R. Socher, X. Giro-i Nieto, and J. Torres.
Explore, discover and learn: Unsupervised discovery of state-covering skills.
In *International Conference on Machine Learning*, 2020.

N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko.
End-to-end object detection with transformers.
In *European Conference on Computer Vision*, 2020.

L. Chen, K. Lu, A. Rajeswaran, K. Lee, A. Grover, M. Laskin, P. Abbeel, A. Srinivas, and I. Mordatch.
Decision transformer: Reinforcement learning via sequence modeling.
arXiv preprint arXiv:2106.01345, 2021.

M. Chen, A. Radford, R. Child, J. Wu, H. Jun, D. Luan, and I. Sutskever.
Generative pretraining from pixels.
In *International Conference on Machine Learning*, pages 1691–1703. PMLR, 2020.

S. Dasari and A. Gupta.
Transformers for one-shot visual imitation.
arXiv preprint arXiv:2011.05970, 2020.

S. Dathathri, A. Madotto, J. Lan, J. Hung, E. Frank, P. Molino, J. Yosinski, and R. Liu.
Plug and play language models: A simple approach to controlled text generation.
arXiv preprint arXiv:1912.02164, 2019.

References vi

J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova.
Bert: Pre-training of deep bidirectional transformers for language understanding.
arXiv preprint arXiv:1810.04805, 2018.

A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al.
An image is worth 16x16 words: Transformers for image recognition at scale.
arXiv preprint arXiv:2010.11929, 2020.

A. Ecoffet, J. Huizinga, J. Lehman, K. O. Stanley, and J. Clune.
Go-explore: A new approach for hard-exploration problems.
arXiv preprint arXiv:1901.10995, 2019.

B. Eysenbach, A. Gupta, J. Ibarz, and S. Levine.
Diversity is all you need: Learning skills without a reward function.
In *International Conference on Learning Representations*, 2019.

J. Ferret, R. Marinier, M. Geist, and O. Pietquin.
Self-attentional credit assignment for transfer in reinforcement learning.
arXiv preprint arXiv:1907.08027, 2019.

J. Ficler and Y. Goldberg.
Controlling linguistic style aspects in neural language generation.
arXiv preprint arXiv:1707.02633, 2017.

J. Fu, A. Kumar, O. Nachum, G. Tucker, and S. Levine.
D4rl: Datasets for deep data-driven reinforcement learning.
arXiv preprint arXiv:2004.07219, 2020.

S. Fujimoto, D. Meger, and D. Precup.
Off-policy deep reinforcement learning without exploration.
In *International Conference on Machine Learning*, 2019.

Y. Gao, H. Xu, J. Lin, F. Yu, S. Levine, and T. Darrell.
Reinforcement learning from imperfect demonstrations.
arXiv preprint arXiv:1802.05313, 2018.

M. Ghazvininejad, X. Shi, J. Priyadarshi, and K. Knight.
Hafez: An interactive poetry generation system.
In *Proceedings of ACL, System Demonstrations*, 2017.

References ix

D. Ghosh, A. Gupta, J. Fu, A. Reddy, C. Devin, B. Eysenbach, and S. Levine.
Learning to reach goals without reinforcement learning.
arXiv preprint arXiv:1912.06088, 2019.

D. Hafner, T. Lillicrap, M. Norouzi, and J. Ba.
Mastering atari with discrete world models.
arXiv preprint arXiv:2010.02193, 2020.

A. Harutyunyan, W. Dabney, T. Mesnard, M. Azar, B. Piot, N. Heess, H. van Hasselt, G. Wayne, S. Singh, and e. a. Precup, Doina.
Hindsight credit assignment.
arXiv preprint arXiv:1912.02503, 2019.

A. Holtzman, J. Buys, M. Forbes, A. Bosselut, D. Golub, and Y. Choi.
Learning to write with cooperative discriminators.
arXiv preprint arXiv:1805.06087, 2018.

Z. Hu, Z. Yang, X. Liang, R. Salakhutdinov, and E. P. Xing.
Toward controlled generation of text.
In *International Conference on Machine Learning*, 2017.

C.-C. Hung, T. Lillicrap, J. Abramson, Y. Wu, M. Mirza, F. Carnevale, A. Ahuja, and G. Wayne.
Optimizing agent behavior over long time scales by transporting value.
Nature Communications, 10(1):1–12, 2019.

M. Janner, J. Fu, M. Zhang, and S. Levine.
When to trust your model: Model-based policy optimization.
In *Advances in Neural Information Processing Systems*, pages 12498–12509, 2019.

M. Janner, Q. Li, and S. Levine.
Reinforcement learning as one big sequence modeling problem.
arXiv preprint arXiv:2106.02039, 2021.

T. Karras, S. Laine, and T. Aila.
A style-based generator architecture for generative adversarial networks.
In *Conference on Computer Vision and Pattern Recognition*, 2019.

N. S. Keskar, B. McCann, L. R. Varshney, C. Xiong, and R. Socher.
Ctrl: A conditional transformer language model for controllable generation.
arXiv preprint arXiv:1909.05858, 2019.

R. Kidambi, A. Rajeswaran, P. Netrapalli, and T. Joachims.
Morel: Model-based offline reinforcement learning.
In *Advances in Neural Information Processing Systems*, 2020.

B. Krause, A. D. Gotmare, B. McCann, N. S. Keskar, S. Joty, R. Socher, and N. F. Rajani.
Gedi: Generative discriminator guided sequence generation.
arXiv preprint arXiv:2009.06367, 2020.

A. Kumar, J. Fu, G. Tucker, and S. Levine.
Stabilizing off-policy q-learning via bootstrapping error reduction.
arXiv preprint arXiv:1906.00949, 2019.

A. Kumar, X. B. Peng, and S. Levine.
Reward-conditioned policies.
arXiv preprint arXiv:1912.13465, 2019.

A. Kumar, A. Zhou, G. Tucker, and S. Levine.
Conservative q-learning for offline reinforcement learning.
In *Advances in Neural Information Processing Systems*, 2020.

S. Levine, A. Kumar, G. Tucker, and J. Fu.
Offline reinforcement learning: Tutorial, review, and perspectives on open problems.
arXiv preprint arXiv:2005.01643, 2020.

Y. Liu, Y. Luo, Y. Zhong, X. Chen, Q. Liu, and J. Peng.
Sequence modeling of temporal credit assignment for episodic reinforcement learning.
arXiv preprint arXiv:1905.13420, 2019.

I. Loshchilov and F. Hutter.
Decoupled weight decay regularization.
arXiv preprint arXiv:1711.05101, 2017.

K. Lu, A. Grover, P. Abbeel, and I. Mordatch.
Reset-free lifelong learning with skill-space planning.
arXiv preprint arXiv:2012.03548, 2020.

T. Mesnard, T. Weber, F. Viola, S. Thakoor, A. Saade, A. Harutyunyan, W. Dabney, T. Stepleton, N. Heess, and e. a. Guez, Arthur.
Counterfactual credit assignment in model-free reinforcement learning.
arXiv preprint arXiv:2011.09464, 2020.

V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller.
Playing atari with deep reinforcement learning.
arXiv preprint arXiv:1312.5602, 2013.
NIPS Deep Learning Workshop 2013.

V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, and e. a. Ostrovski, Georg.
Human-level control through deep reinforcement learning.
Nature, 518(7540):529–533, 2015.

A. Nair, M. Dalal, A. Gupta, and S. Levine.
Accelerating online reinforcement learning with offline datasets.
arXiv preprint arXiv:2006.09359, 2020.

E. Parisotto, F. Song, J. Rae, R. Pascanu, C. Gulcehre, S. Jayakumar, M. Jaderberg, R. Lopez Kaufman, A. Clark, and e. a. Noury, Seb.
Stabilizing transformers for reinforcement learning.
In *International Conference on Machine Learning*, 2020.

K. Paster, S. A. McIlraith, and J. Ba.
Planning from pixels using inverse dynamics models.
arXiv preprint arXiv:2012.02419, 2020.

X. B. Peng, A. Kumar, G. Zhang, and S. Levine.
Advantage-weighted regression: Simple and scalable off-policy reinforcement learning.
arXiv preprint arXiv:1910.00177, 2019.

K. Pertsch, Y. Lee, and J. J. Lim.
Accelerating reinforcement learning with learned skill priors.
arXiv preprint arXiv:2010.11944, 2020.

A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever.
Improving language understanding by generative pre-training.
2018.

N. F. Rajani, B. McCann, C. Xiong, and R. Socher.
Explain yourself! leveraging language models for commonsense reasoning.
arXiv preprint arXiv:1906.02361, 2019.

A. Ramesh, M. Pavlov, G. Goh, S. Gray, C. Voss, A. Radford, M. Chen, and I. Sutskever.
Zero-shot text-to-image generation.
arXiv preprint arXiv:2102.12092, 2021.

D. Raposo, S. Ritter, A. Santoro, G. Wayne, T. Weber, M. Botvinick, H. van Hasselt, and F. Song.
Synthetic returns for long-term credit assignment.
arXiv preprint arXiv:2102.12425, 2021.

S. Ritter, R. Faulkner, L. Sartran, A. Santoro, M. Botvinick, and D. Raposo.
Rapid task-solving in novel environments.
arXiv preprint arXiv:2006.03662, 2020.

A. Sharma, S. Gu, S. Levine, V. Kumar, and K. Hausman.
Dynamics-aware unsupervised discovery of skills.
In *International Conference on Learning Representations*, 2020.

N. Y. Siegel, J. T. Springenberg, F. Berkenkamp, A. Abdolmaleki, M. Neunert, T. Lampe, R. Hafner, and M. Riedmiller.
Keep doing what worked: Behavioral modelling priors for offline reinforcement learning.
In *International Conference on Learning Representations*, 2020.

References xx

A. Singh, H. Liu, G. Zhou, A. Yu, N. Rhinehart, and S. Levine.
Parrot: Data-driven behavioral priors for reinforcement learning.
In *International Conference on Learning Representations*, 2021.

R. K. Srivastava, P. Shyam, F. Mutz, W. Jaskowski, and J. Schmidhuber.
Training agents using upside-down reinforcement learning.
arXiv preprint arXiv:1912.02877, 2019.

R. S. Sutton.
Integrated architectures for learning, planning, and reacting based on approximating dynamic programming.
In *ICML*, 1990.

R. S. Sutton and A. G. Barto.
Reinforcement Learning: An Introduction.
The MIT Press, 2018.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin.
Attention is all you need.
In *Advances in Neural Information Processing Systems*, 2017.

C. Watkins and P. Dayan.
Q-learning.
Machine Learning, 8:279–292, 1992.

L. Weng.
Controllable neural text generation, 2021.

- T. Wolf, J. Chaumond, L. Debut, V. Sanh, C. Delangue, A. Moi, P. Cistac, M. Funtowicz, J. Davison, S. Shleifer, et al.
  Transformers: State-of-the-art natural language processing.
  In *Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 2020.
- Y. Wu, G. Tucker, and O. Nachum.
  Behavior regularized offline reinforcement learning.
  arXiv preprint arXiv:1911.11361, 2019.
- L. Yu, W. Zhang, J. Wang, and Y. Yu.
  Seqgan: Sequence generative adversarial nets with policy gradient.
  In *AAAI Conference on Artificial Intelligence*, 2017.

- T. Yu, G. Thomas, L. Yu, S. Ermon, J. Zou, S. Levine, C. Finn, and T. Ma.
  Mopo: Model-based offline policy optimization.
  In *Advances in Neural Information Processing Systems*, 2020.
- V. Zambaldi, D. Raposo, A. Santoro, V. Bapst, Y. Li, I. Babuschkin, K. Tuyls, D. Reichert, T. Lillicrap, and e. a. Lockhart, Edward.
  Deep reinforcement learning with relational inductive biases.
  In *International Conference on Learning Representations*, 2018.
- D. M. Ziegler, N. Stiennon, J. Wu, T. B. Brown, A. Radford, D. Amodei, P. Christiano, and G. Irving.
  Fine-tuning language models from human preferences.
  arXiv preprint arXiv:1909.08593, 2019.