Body Transformer: Leveraging Robot Embodiment for Policy Learning

Anonymous Author(s)
Affiliation
Address
email

Abstract: In recent years, the transformer architecture has become the de-facto standard for machine learning algorithms applied to natural language processing and computer vision. Despite notable evidence of successful deployment of this architecture in the context of robot learning, we claim that vanilla transformers do not fully exploit the structure of the robot learning problem. We propose Body Transformer (BoT), an architecture that exploits the robot embodiment by providing an inductive bias that guides the learning process. We represent the robot body as a graph of sensors and actuators, and rely on masked attention to pool information through the architecture. The resulting architecture outperforms the vanilla transformer, as well as the classical multilayer perceptron, with respect to task completion, scaling properties, and computational efficiency when representing either imitation or reinforcement learning policies. Additional material is available at bodytransformer.site. Code will be open-sourced upon acceptance.

Keywords: Robot Learning, Graph Neural Networks, Imitation Learning, Reinforcement Learning

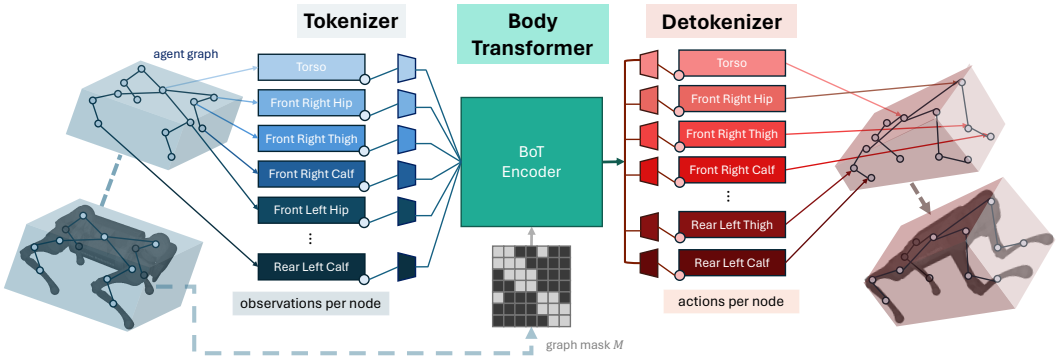


Figure 1: **Body Transformer** (BoT) is an architecture that considers physical agents as graphs of sensors and actuators as nodes, and edges reflecting the structure of the robot body. BoT leverages masked self-attention as a simple but flexible mechanism to provide a body-induced bias to the policy. The figure presents the overall schematic of our architecture, exemplified on a Unitree A1 robot.

1 Introduction

For most of their correcting and stabilizing actions, physical agents exhibit motor responses that are spatially correlated with the location of the external stimuli they perceive [1]. This is the case of a surfer, where the lower body, i.e. feet and ankles, is mostly responsible for counteracting the

Submitted to the 8th Conference on Robot Learning (CoRL 2024). Do not distribute.

imbalance induced by the wave under the board [2]. In fact, humans present feedback loops at the level of the spinal cord's neural circuitry that are specifically responsible for the response of single actuators [3].

Corrective localized actuation is a main factor for efficient locomotion [4]. This is particularly important for robots too, where however, learning architectures do not typically exploit spatial inter-relations between sensors and actuators. In fact, robot policies have mostly been exploiting the same architectures developed for natural language or computer vision, without effectively leveraging the structure of the robot body.

This work focuses on transformer policies, which show promise to effectively deal with long sequence dependencies and seamlessly absorb large amount of data. The transformer architecture [5] has been developed for unstructured natural language processing (NLP) tasks, e.g., language translations, where the input sequences often map to reshuffled output sequences. In contrast, we propose Body Transformer (BoT), an architecture that augments the attention mechanism of transformers by taking into account the spatial placement of sensors and actuators across the robot body.

BoT models the robot body as a graph with sensors and actuators at its nodes. Then, it applies a highly sparse mask at the attention layers, preventing each node from attending beyond its direct neighbors. Concatenating multiple BoT layers with the same structure leads to information being pooled throughout the graph, thus not compromising the representation power of the architecture.

Our contributions are listed below:

- We propose the BoT architecture, which augments the transformer architecture with a novel masking that leverages the morphology of the robot body.
- We incorporate this novel architecture in an imitation learning setting, showing how the inductive bias provided by BoT leads to better steady-state performance and generalization, as well as better scaling properties.
- We show how BoT improves online reinforcement learning (RL), outperforming MLP and vanilla transformer baselines.
- We analyze the computational advantages of BoT, by showing how reformulating the scaled dot product in the computation of the attention operation leads to near-200% runtime and floating point operations (FLOPs) reduction.

2 Related Work

Transformers in robotics. Originally developed for NLP applications [5], transformers have been successfully applied across domains, for example in computer vision [6] and audio processing [7]. Several works have shown applications of transformers as a means to represent robot policies [8, 9, 10], demonstrating its core advantages in this setting, i.e., variable context length, handling long sequences [11] and multiple modalities [12, 13]. However, these approaches use transformers as originally developed for unstructured or grid-like inputs, such as language or images, respectively. In this work, we leverage the robot embodiment by appropriately adapting the transformer attention mechanism.

Graph Neural Networks (GNNs). GNNs [14] are a class of learning architectures that can process inputs in the form of a graph [15]. While early versions of these architectures featured explicit message-passing schemes along the graph [16, 17], more recent architectures mostly feature attention-based approaches. In fact, the vanilla transformer, with its variable context length, inherently supports fully connected graphs. However, state-of-the-art performance on graph interpretation benchmarks is only achieved via modifications of the original transformer architecture (eg., learned graph encodings and attention biases [18]). A contemporaneous work, Buterez et al. [19], following a similar idea as in the work from Veličković et al. [20], introduces masked attention layers, where each node only attends to its neighbors, and interleaves such layers with unmasked attention layers. In this work, we propose an architecture that only comprises layers where each can attend to itself

and its direct neighbors, resulting in naturally growing context over layers, i.e., the outputs of the first layers are computed using more local information compared to those of the last layers.

Exploiting body structure for policy learning. Graph neural networks have been explored by several works as a way to obtain multi-task RL policies that are effective across different robot morphologies. Earlier works focused on message passing algorithms [21, 22], and were later outperformed by vanilla transformers [23] and transformer-based GNNs that make use of learned encoding and attention biases [24]. Compared to previous work, we show that introducing bottlenecks in the attention mask fully exploits the embodiment structure and benefits policy learning also for tasks achieved by a single agent, leading to better performance and more favorable scaling.

3 Background

3.1 Attention Mechanisms in Transformers

Transformer, a foundational architecture in modern machine learning applications as well as in our work, is powered by the self-attention mechanism [5]. Self-attention weighs the values corresponding to each element of the sequence with a score that is computed from pairs of keys and queries extracted from the same sequence. Thus, it is able to identify relevant pairs of sequence elements in the model output.

Concretely, the self-attention output vector is computed through the following matrix operation:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V,$$

where Q, K, V (respectively query, key, and value matrices) are learnable linear projections of the sequence elements' embedding vectors, and d_k is the dimension of the key vector. As embedding pairs with higher correspondence will have a higher score (from the computation of QK^T), the corresponding value vector of an embedding's associated key will receive a higher weight in the attention mechanism output.

3.2 Transformer-based GNNs

In this work, we model the agent embodiment as a graph whose nodes are sensors and actuators, and their connecting edges reflect the body morphology. While message-passing GNNs are suitable inductive biases for this formulation, they tend to suffer from oversmoothing and oversquashing of representations, preventing effective long-range interactions and discouraging network depth [25].

More recently, self-attention was proposed as an alternative to message-passing [18, 23]. While the standard self-attention mechanism amounts to modeling a fully connected graph, a popular transformer-based GNN, Graphormer [18], injects node-edges information through graph-based positional encodings [26, 27, 18, 24] and by biasing the scaled dot-product–

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}} + B\right)V, \tag{1}$$

B is a learnable matrix that depends on graph features, e.g., shortest path or adjacency matrix, and can effectively encode the graph structure in a flexible manner.

3.3 Masked Attention

The attention mechanism can be altered [5] with a binary mask $M \in \{0, 1\}^{n \times n}$ (where n is the sequence length), which is equivalent to replacing the elements of B in (1):

$$B_{i,j} = \begin{cases} 0 & M_{i,j} = 1 \\ -\infty & M_{i,j} = 0 \end{cases},$$

where i and j denote row and column indices. This operation effectively results in zeroing out the contribution of the pairs indicated with zeros in the mask M.

4 Body Transformer

Robot learning policies that employ the vanilla transformer architecture as a backbone typically neglect the useful information provided by the embodiment structure. In contrast, here we leverage this structure to provide a stronger inductive bias to transformers, while retaining the representation power of the original architecture. We propose Body Transformer (BoT), which is based on masked attention, where at each layer in the resulting architecture, a node can only attend to information from itself and its direct neighbors. As a result, information flows according to the graph structure, with the upstream layers reasoning according to local information and the downstream layers pooling more global information from the farther nodes.

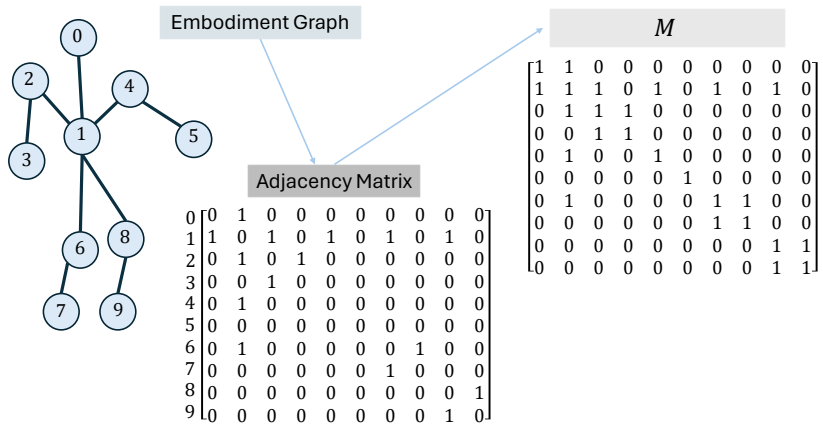


Figure 2: **Formulation of Mask M.** The mask M is formulated by adding a diagonal of 1s to the embodiment graph's adjacency matrices. Here, we visualize a simple example of formulating a mask M for an arbitrary agent's embodiment where $n = 10$.

We present below the various components of the BoT architecture: (1) a tokenizer that projects the sensory inputs into the corresponding node embedding, (2) a transformer encoder that processes the input embeddings and generates output features of the same dimension, and (3) a detokenizer that decodes the features to actions (or values, for RL training).

Tokenizer. We map the observation vector to a graph of local observations. In practice, we assign global quantities to the root element of the body, and local quantities to the nodes representing the corresponding limbs, similarly to previous GNN approaches Huang et al. [22], Kurin et al. [23], Hong et al. [24]. Then, a linear layer projects the local state vector into an embedding vector. Each node's state is fed into its *node-specific* learnable linear projection, resulting in a sequence of n embeddings, where n represents the number of nodes (or sequence length). This is in contrast to the existing works [23, 24], where they use a single *shared* learnable linear projection to deal with varying number of nodes in multi-task RL.

BoT Encoder. We use a standard transformer encoder [5] with several layers as a backbone, and present two variants of our architecture:

- BoT-Hard masks every layer with a binary mask M that reflects the structure of the graph. Specifically, we construct the mask as $M = I_n + A$, where I_n is the identity matrix of dimension n, and A is the adjacency matrix corresponding to the graph (see Figure 2 for an example). Concretely, this allows each node to attend to itself and its direct neighbors, and introduce considerable sparsity in the problem, which is particularly appealing from a computational perspective as highlighted in Section 5.4.
- BoT-Mix interleaves layers with masked attention (constructed as in BoT-Hard) with layers with unmasked attention. This is similar to the concurrent work in Buterez et al. [19], with the distinctions that, I) we find it more effective in our experimental setting to have a

masked attention layer as the first layer, II) our mask M is not equivalent to the adjacency matrix, allowing a node to attend to itself at every layer of the architecture.

Detokenizer. The output features from the transformer encoder are fed into linear layers that project them to the actions associated with the node's limb, which are assigned based on the proximity of the corresponding actuator with the limb. Once again, these learnable linear projection layers are separate for each of the nodes. When BoT is employed as a critic architecture in the RL setting, as in the experiments presented below, the detokenizer outputs values rather than actions.

5 Experiments

We assess the performance of BoT across imitation learning and reinforcement learning settings. We keep the same structure as in Figure 1 and only replace the BoT encoder with the various baseline architectures. Particularly, across the various experiments recorded in this section, we present the following baselines and variations: (1) an MLP that stacks all embedding vectors as its input, (2) a vanilla unmasked transformer encoder, (3) BoT-Hard that only uses masked self-attention layers, (4) BoT-Mix that alternates between masked and unmasked self-attention layers. All comparisons are made across models with a similar number of trainable parameters.

With the following experiments, we aim to answer the following questions:

- Does masked attention benefit imitation learning in terms of performance and generalization?
- Does BoT exhibit a positive scaling trend compared to a vanilla transformer architecture?
- Is BoT compatible with the RL framework, and what are sensible design choices to maximize performance?
- Can BoT policies be applied to a realistic robotic task?
- What are the computational advantages of masked attention?

5.1 Imitation Learning Experiments

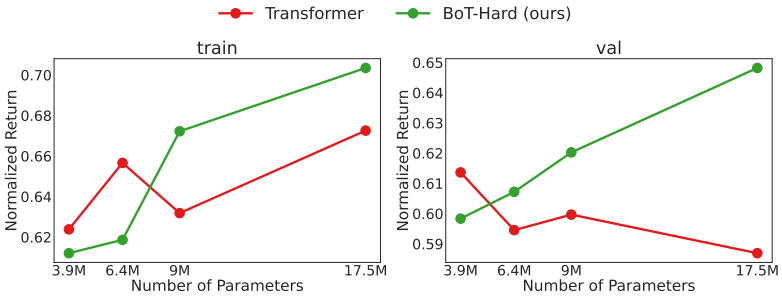
We evaluate the imitation learning performance of the BoT architecture in a body-tracking task defined through the MoCapAct dataset [28], which comprises action-labeled humanoid state trajectories with over 5M transitions, spanning a total of 835 tracking clips. For each architecture, we train a behavioral cloning (BC) deterministic policy. We evaluate mean returns normalized by the length of a clip, in addition to the normalized length of an episode, which terminates when the tracking error goes beyond a threshold. We run the evaluations both on the training and the (unseen) validation clips.

We report results in the table shown in Figure 3a, where BoT consistently outperforms the MLP and transformer baselines. Remarkably, the gap with these architectures further increases on the unseen validation clips, demonstrating the generalization capabilities provided by the embodiment-aware inductive bias. We also report the performance on a training set obtained by a tailored multi-clip policy presented in Wagener et al. [28] with the MoCapAct dataset. While the multi-clip policy is competitive with the vanilla transformer baseline, it is strongly outperformed by our architecture. This is a particularly remarkable result, as the comparison presents conditions more favorable to the baseline, which features a more flexible stochastic policy, was optimized in a recurrent fashion tailored to the tracking task, and was trained on a larger set of rollouts.

As shown in Figure 3b, we also find that BoT-Hard exhibits strong scaling capabilities, as its performance keeps improving with the number of trainable parameters compared to the transformer baseline, both on the training and validation clips. This further indicates a tendency for BoT-Hard to not overfit to the training data, which is induced by the embodiment bias. Additional experiments and comparisons are reported in the appendix.

	normalized episode return		normalized episode length	
	train	val	train	val
MLP	0.821 / 0.586	0.739 / 0.540	0.957 / 0.765	0.915 / 0.726
Transformer	**0.909** / 0.673	0.806 / 0.587	0.995 / 0.855	0.952 / 0.776
BoT-Hard (ours)	0.908 / **0.703**	**0.890 / 0.648**	**1.000 / 0.876**	**1.000 / 0.841**
Multi-Clip [28]	- / 0.654	- / -	- / 0.855	- / -

(a) **Training, Testing Performance Across Architectures.** Statistics of the same architecture-criterion combination are shown with two values, the leftside being the maximum value achieved throughout three trajectories, the rightside being the mean of all trajectories' last 15 logged eval epochs.



(b) **Training, Testing Performance Across Number of Parameters.** Each datapoint representing a parameter count-architecture combination is the mean of its trajectories' last 15 logged eval epochs. We use the 17.5M models for the other imitation learning experiments in this paper.

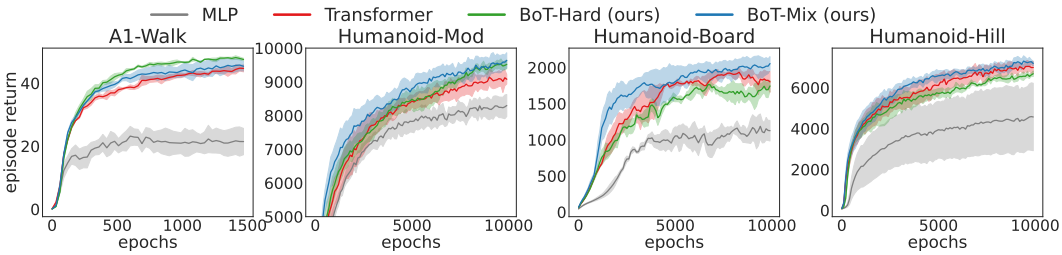
Figure 3: **BoT Performance on Imitation Learning.**

5.2 Reinforcement Learning Experiments

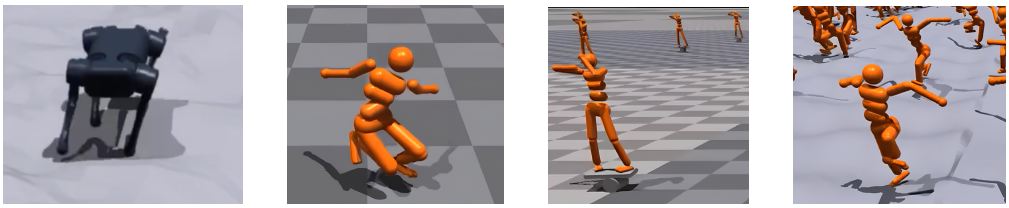
We evaluate the RL performance of BoT and baselines on 4 robotic control tasks in Isaac Gym [29]: Humanoid-Mod, Humanoid-Board, Humanoid-Hill, and A1-Walk.

All humanoid environments build on top of the classical Humanoid environment in Isaac Gym, where we modify the observation space to increase the number of distributed sensory information (see details in the appendix) and include contact forces at all limbs. Humanoid-Mod features the classical running task on flat ground, while in Humanoid-Hill we replaced the flat ground with an irregular hilly terrain. Humanoid-Board is a newly designed task, where the task is for the humanoid to keep balancing on a board placed on top of a cylinder. Finally, we adapt the A1-Walk environment, which is part of the Legged Gym repository [30], where the task is for a Unitree A1 quadruped robot to follow a fixed velocity command.

Figure 4 presents the average episode return of evaluation rollouts during training for MLP, Transformer, and BoT (Hard and Mix). The solid curve corresponds to the mean, and the shaded area to the standard error over three seeds. The result shows that BoT-Mix consistently outperforms both the MLP and vanilla transformer baselines in terms of sample efficiency and asymptotic performance, highlighting the efficacy of integrating body-induced bias into the policy network architecture. Meanwhile, BoT-Hard performs better than the vanilla transformer on simpler tasks (A1-Walk and Humanoid-Mod), but shows relatively inferior results in hard-exploration tasks (Humanoid-Board and Humanoid-Hill). Given that the masked attention bottlenecks information propagation from distant body parts, BoT-Hard's strong constraints on information communication between bodies may hinder efficient RL exploration: In Humanoid-Board and Humanoid-Hill, it may be useful for information about sudden changes in ground conditions to be transmitted from the toes to the fingertips in the upstream layers. For such tasks, BoT-Mix finds a good balance between funneling information through the embodiment graph and enabling global pooling at intermediate layers to warrant efficient exploration. In contrast, in A1-Walk or Humanoid-Mod, the environment's state changes more regularly, thus the strong body-induced bias can effectively reduce the search space, enabling faster learning with BoT-Hard.



(a) Performance on various tasks across evaluated architectures.



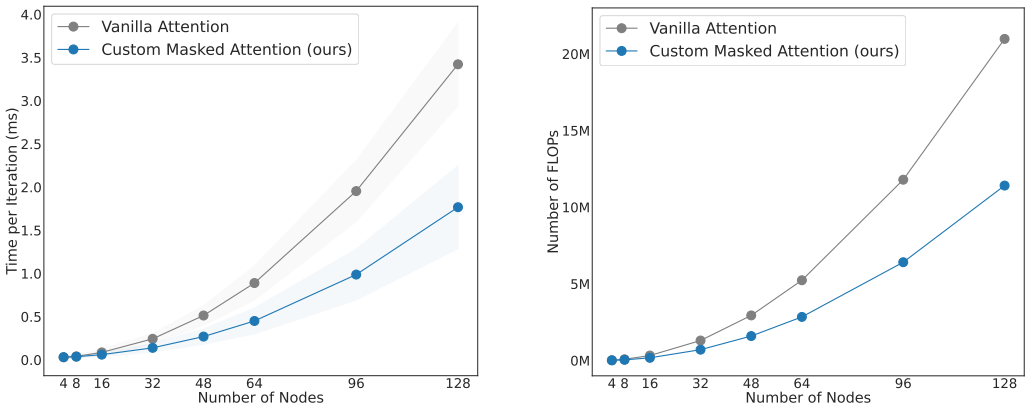
(b) Captured images of success in tasks from BoT policies.

Figure 4: **Reinforcement Learning Performance on Robotic Control Tasks.**

5.3 Real World Experiments

The Isaac Gym simulated locomotion environments are a widely popular for sim-to-real transfer of RL policies without adaptation in the real-world. To verify that our architecture is suitable for real-world applications, e.g., running in real time, we deploy one of the BoT policies trained above to a real-world Unitree A1 robot. This is showcased in the supplementary video, demonstrating feasibility of our architecture for real-world deployment. We note that for simplicity we did not make use of teacher-student training or memory mechanisms as common in the locomotion literature, which are known to further improve the transfer by resulting in more natural gaits.

5.4 Computational Analysis



(a) Runtime

(b) FLOPs

Figure 5: **Computational analysis of the custom masked attention.** Across 10,000 randomized masks, we found our sparsified implementation already providing a 200% speedup in runtime at small sequence lengths. Our custom implementation also scales better in number of FLOPs.

Connections between body parts of a physical agent are often sparse, and so would the pre-computable masks M for embodiment graphs in BoT. Masked attention mechanisms can benefit from this sparsity, as their computational cost can be reduced by ignoring unnecessary computa-

tions of those matrix elements that will eventually be masked out. Large-purpose deep learning libraries such as `PyTorch` feature largely optimized matrix multiplication and attention routines (e.g., FlashAttention [31]), but do not leverage possible sparsity in the masked attention mechanism, ascribed by Buterez et al. [19] to missing use cases so far. For a fair computational comparison, we re-implement the soft dot product in Equation (1) using CPU-based `NumPy` and evaluate on a single sample and attention head, being their batched and multi-head versions further parallelizable on GPUs.

Our custom implementation comprises of two larger changes to the computation of the masked attention mechanism from its vanilla implementation in `PyTorch`. First of all, we only perform the dot product computation for elements that will not be masked, resulting in fewer matrix multiplication-induced FLOPs from the computation of $\frac{1}{\sqrt{d_k}}QK^T$. Secondly, we only use unmasked values to compute the softmax in Equation (1), also resulting in reduced FLOPs.

We measure the average runtime of each implementation of the attention mechanism across 10,000 set of randomly generated Q, K, V, and M. For each randomization, the generated masks M have a diagonal of 1s and sparsity equal to that used in the MoCapAct experiments ($\beta = 0.908$) [1].

In Figure 5a, we present scaling results of our implementation against vanilla attention across sequence lengths (number of nodes) $\{4, 8, 16, 32, 48, 64, 96, 128\}$. We observe potential speedups of up to 206% for number of nodes up to 128 (e.g., in the order of humanoids with dexterous hands [32]). Overall, this shows that the body-induced bias in our BoT architecture not only enhances the overall performance of physical agents but also benefits from the naturally sparse masks that the architecture imposes. With adequate parallelization, this approach mat significantly reduce the training time of learning algorithms as shown above.

A depiction of the amount of FLOPs required across different sequence lengths is also presented in Figure 5b. Further details and derivation about these experiments can be found in the appendix.

6 Conclusion

In this work, we presented a novel graph-based policy architecture, Body Transformer, which exploits the robot body structure as an inductive bias. Our experiments show how masked attention, which is at the core of BoT, benefits both imitation and reinforcement learning algorithms. Additionally, the architecture exhibits favorable scaling and computational properties, making it appealing for applications on high-dimensional systems.

Here, we used transformers to process sequences of distributed sensory information from the same timestep. However, transformers have been shown to excel at processing information across time too. We leave the extension of BoT to the temporal dimension as future work, as it promises to further improve real world deployment of robot policies, such as the one demonstrated on the Unitree A1 robot.

A limitation of our approach is the fact that its computational advantages are currently not leveraged by modern deep learning libraries, and we hope that this work may stimulate future developments in this direction. In addition, our architecture requires a minimum amount of transformer layers to make sure that the architecture does not lose representation power in modeling long-range relations, which generally increases the amount of required trainable parameters.

[1] A mask with sparsity β has βn^2 zero elements. When $\beta = 1 - \frac{1}{n}$, the mask reduces to the identity I_n. In practice, the maximum degree of nodes in robots would be constant, where the computational complexity of masked attention will grow linearly to the number of nodes.

References

[1] H. Forssberg. Stumbling corrective reaction: a phase-dependent compensatory reaction during locomotion. *Journal of neurophysiology*, 42(4):936–953, 1979.

[2] J. L. Secomb, O. R. Farley, L. Lundgren, T. T. Tran, A. King, S. Nimphius, and J. M. Sheppard. Associations between the performance of scoring manoeuvres and lower-body strength and power in elite surfers. *International Journal of Sports Science & Coaching*, 10(5):911–918, 2015.

[3] L. Seminara, S. Dosen, F. Mastrogiovanni, M. Bianchi, S. Watt, P. Beckerle, T. Nanayakkara, K. Drewing, A. Moscatelli, R. L. Klatzky, et al. A hierarchical sensorimotor control framework for human-in-the-loop robotic hands. *Science Robotics*, 8(78):eadd5434, 2023.

[4] S. H. Collins and A. D. Kuo. Recycling energy to restore impaired ankle function during human walking. *PLoS one*, 5(2):e9307, 2010.

[5] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[6] 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.

[7] L. Dong, S. Xu, and B. Xu. Speech-transformer: a no-recurrence sequence-to-sequence model for speech recognition. In *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 5884–5888. IEEE, 2018.

[8] T. Z. Zhao, V. Kumar, S. Levine, and C. Finn. Learning fine-grained bimanual manipulation with low-cost hardware. *arXiv preprint arXiv:2304.13705*, 2023.

[9] 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. *Advances in neural information processing systems*, 34:15084–15097, 2021.

[10] Q. Zheng, A. Zhang, and A. Grover. Online decision transformer. In *international conference on machine learning*, pages 27042–27059. PMLR, 2022.

[11] I. Radosavovic, B. Zhang, B. Shi, J. Rajasegaran, S. Kamat, T. Darrell, K. Sreenath, and J. Malik. Humanoid locomotion as next token prediction. *arXiv preprint arXiv:2402.19469*, 2024.

[12] H. Qi, B. Yi, S. Suresh, M. Lambeta, Y. Ma, R. Calandra, and J. Malik. General in-hand object rotation with vision and touch. In *Conference on Robot Learning*, pages 2549–2564. PMLR, 2023.

[13] C. Sferrazza, Y. Seo, H. Liu, Y. Lee, and P. Abbeel. The power of the senses: Generalizable manipulation from vision and touch through masked multimodal learning, 2023.

[14] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and S. Y. Philip. A comprehensive survey on graph neural networks. *IEEE transactions on neural networks and learning systems*, 32(1): 4–24, 2020.

[15] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini. The graph neural network model. *IEEE transactions on neural networks*, 20(1):61–80, 2008.

[16] P. W. Battaglia, J. B. Hamrick, V. Bapst, A. Sanchez-Gonzalez, V. Zambaldi, M. Malinowski, A. Tacchetti, D. Raposo, A. Santoro, R. Faulkner, et al. Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261*, 2018.

[17] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl. Neural message passing for quantum chemistry. In *International conference on machine learning*, pages 1263–1272. PMLR, 2017.

[18] C. Ying, T. Cai, S. Luo, S. Zheng, G. Ke, D. He, Y. Shen, and T.-Y. Liu. Do transformers really perform badly for graph representation? *Advances in neural information processing systems*, 34:28877–28888, 2021.

[19] D. Buterez, J. P. Janet, D. Oglic, and P. Lio. Masked attention is all you need for graphs, 2024.

[20] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017.

[21] T. Wang, R. Liao, J. Ba, and S. Fidler. Nervenet: Learning structured policy with graph neural networks. In *International conference on learning representations*, 2018.

[22] W. Huang, I. Mordatch, and D. Pathak. One policy to control them all: Shared modular policies for agent-agnostic control. In *International Conference on Machine Learning*, pages 4455–4464. PMLR, 2020.

[23] V. Kurin, M. Igl, T. Rocktäschel, W. Boehmer, and S. Whiteson. My body is a cage: the role of morphology in graph-based incompatible control, 2021.

[24] S. Hong, D. Yoon, and K.-E. Kim. Structure-aware transformer policy for inhomogeneous multi-task reinforcement learning. In *International Conference on Learning Representations*, 2021.

[25] U. Alon and E. Yahav. On the bottleneck of graph neural networks and its practical implications, 2021.

[26] W. Park, W. Chang, D. Lee, J. Kim, and S. won Hwang. Grpe: Relative positional encoding for graph transformer, 2022.

[27] J. Yeom, T. Kim, R. Chang, and K. Song. Structural and positional ensembled encoding for graph transformer. *Pattern Recognition Letters*, 2024.

[28] N. Wagener, A. Kolobov, F. V. Frujeri, R. Loynd, C.-A. Cheng, and M. Hausknecht. Mocapact: A multi-task dataset for simulated humanoid control, 2023.

[29] V. Makoviychuk, L. Wawrzyniak, Y. Guo, M. Lu, K. Storey, M. Macklin, D. Hoeller, N. Rudin, A. Allshire, A. Handa, and G. State. Isaac gym: High performance gpu-based physics simulation for robot learning, 2021.

[30] N. Rudin, D. Hoeller, P. Reist, and M. Hutter. Learning to walk in minutes using massively parallel deep reinforcement learning. In *Conference on Robot Learning*, pages 91–100. PMLR, 2022.

[31] T. Dao, D. Y. Fu, S. Ermon, A. Rudra, and C. Ré. Flashattention: Fast and memory-efficient exact attention with io-awareness, 2022.

[32] C. Sferrazza, D.-M. Huang, X. Lin, Y. Lee, and P. Abbeel. Humanoidbench: Simulated humanoid benchmark for whole-body locomotion and manipulation. *arXiv Preprint arxiv:2403.10506*, 2024.