MORE-3S:Multimodal-based Offline Reinforcement Learning with Shared Semantic Spaces

Tianyu Zheng[1]   Ge Zhang[2,3†]   Xingwei Qu[3]
Ming Kuang[4]   Stephen W. Huang[5]   Zhaofeng He[1†]
[1]Beijing University of Posts and Telecommunications   [2]University of Waterloo
[3]Multimodal Art Projection Research Community   [4]ClouDr   [5]Harmony.ai

Abstract

Drawing upon the intuition that aligning different modalities to the same semantic embedding space would allow models to understand states and actions more easily, we propose a new perspective to the offline reinforcement learning (**RL**) challenge. More concretely, we transform it into a supervised learning task by integrating multimodal and pre-trained language models. Our approach incorporates state information derived from images and action-related data obtained from text, thereby bolstering RL training performance and promoting long-term strategic thinking. We emphasize the contextual understanding of language and demonstrate how decision-making in RL can benefit from aligning states' and actions' representation with languages' representation. Our method significantly outperforms current baselines as evidenced by evaluations conducted on Atari and OpenAI Gym environments. This contributes to advancing offline RL performance and efficiency while providing a novel perspective on offline RL.

1. Introduction

RL operates as a sequential process where an agent perceives an environmental state, executes an action, observes the subsequent state, and receives a reward. Offline RL mitigates the limitations of conventional RL, like the need for long-term credit assignment bootstrapping and undesirable short-sighted learned policies, by learning policies from pre-existing datasets (Chen et al., 2021), eliminating the need for environmental interaction (Levine et al., 2020). This paradigm has attracted significant attention for its practical application potential in real-world RL training scenarios.

Unlike conventional offline RL techniques that primarily focus on learning value functions (Kostrikov et al., 2021) or policy gradients (Fujimoto and Gu, 2021), recent approaches (Chen et al., 2021; Schmied et al., 2023) view offline RL as a sequence modeling task. In these methods, past experiences of state-action-reward triplets are input into a transformer model (Vaswani et al., 2017). They create a sequence of anticipated actions through a goal-conditioned policy, casting offline RL as a supervised learning problem. These approaches break away from the MDP assumption by using historical information to predict actions, effectively handling extensive sequences, and ensuring stability, which circumvents bootstrapping-related issues (Kumar et al., 2020).

However, it remains unclear how to effectively adapt Large-scale Pretrained Models (**LPM**) for offline RL Chen et al. (2021). Reid et al. (2022); Zhang et al. (2023) demonstrate the effectiveness

of fine-tuning LPM for offline RL tasks requiring high-level planning. Although further empirical evidence supports these findings (Liu et al., 2022a; Fan et al., 2022), the reasons behind these enhancements largely remain unexplored.

To understand the underlying factors, we suggest two main considerations: first, the *action rules*, such as grammatical structures or pragmatic rules essential for constructing valid sentences; second, the *use cases*, implying the context where words and idioms are applied, especially when considering language as a complex game.

Within the RL and Natural Language Processing (**NLP**) contexts, we suggest that the latent representation of language may significantly contribute to RL decision-making processes. This interpretation views the next token prediction procedure in training Pre-trained Language Models (**PLMs**) as agents continuously making decisions, ascribing meaning to decisions through use. Therefore, integrating PLMs with RL may enhance an agent's ability to interpret and act meaningfully within its environment. Despite its theoretical appeal, empirical evidence supporting this hypothesis is sparse. A persistent challenge is decoupling high-level planning from low-level representations in the models (Correia and Alexandre, 2022). While preliminary attempts (Lee et al., 2022) show promising improvements based on decoupling, there is a pressing need for further research to unravel the mechanisms behind this phenomenon and to develop more effective methods to achieve it.

This study argues that the latent state representations derived from images during offline RL training, coupled with the discrete symbolic action

---

† Corresponding Author.

space that models game operations, should correlate with their corresponding textual descriptions. Chen et al. (2021); Reid et al. (2022) directly utilize the latent state representations and discrete symbolic action space, without establishing alignment with their textual descriptions.

Consider the game "Pong." In a specific frame, the left paddle is close to the top, the ball moves to the left, and the right paddle rises. This can be semantically described as: *"Left paddle near top; ball moving left; right paddle rising"*. Traditional multimodal methods, which lack semantic grounding, might correlate raw pixels with actions like "move paddle up." However, the absence of a higher-level contextual understanding may reduce their resilience to minor game variations. Grounding in a semantic space provides increased adaptability and learning efficiency.

Existing online RL methods (Wang et al., 2023b) usually seek to adapt LPM for high-level RL planning, introducing text-described states empirically without quantifying the advantage of modality alignment. This alignment allows the LPM to understand states and actions better. Nevertheless, the alignment of latent state representations, action space, and their textual descriptions facilitates agents initialized with LPMs' parameters to comprehend states and actions more effectively. As a result, we propose a novel approach integrating multimodal and pre-trained sequence models into sequential RL, as shown in Fig. 1. We design **M**ultimodal **O**ffline **R**einforcement l**E**arning with **S**hared **S**emantic **S**paces (**MORE-3S**), which demonstrates strong performance across various benchmark environments and offers improvements over existing offline RL methods. In summary, our primary contributions are three-fold:

- We introduce a novel approach that aligns multimodal models with pre-trained sequence models for sequential RL, thereby enhancing RL training performance.

- MORE-3S, aligning states' and actions' representation with PLM's latent space, significantly improves the performance of offline RL algorithm. MORE-3S's performance is another convincing empirical observation proving the advances in high-level planning observed where NLP models are utilized in RL.

- We propose integrating the *return-to-go* into the attention mechanism of decision transformer-based RL models, enhancing their utilization of return-to-go.

2. Related Work

Offline Reinforcement Learning   Offline RL, a rapidly expanding field, focuses on deriving opti-

mal policies from fixed datasets of trajectory rollouts, negating the need for further environmental interaction. These datasets, represented by $D = (s_t, a_t, r_t)_{t=0}^{T}$, encompass states ($s_t$), actions ($a_t$), and rewards ($r_t$) for each timestep ($t$) until the episode time horizon (T), and are the products of a behavior policy and inherent dynamics. Offline RL, which utilizes value-based and model-based strategies, boasts a collection of noteworthy algorithms including BCQ (Fujimoto et al., 2019), AWR (Peng et al., 2019), BRAC (Wu et al., 2019), ICQ (Yang et al., 2021), CQL (Kumar et al., 2020), UWAC (Wu et al., 2021), MOPO (Yu et al., 2020), MOReL (Kidambi et al., 2020), Decision Transformer(DT)(Chen et al., 2021), and Trajectory Transformer (Janner et al., 2021). Our approach innovatively integrates multimodal and pre-trained sequence models with sequential RL, aiming to overcome the limitations inherent in fixed-dataset policy optimization.

Sequence Modeling in RL   Research interest in the application of sequence modeling in RL has surged due to the use of Transformer-based decision models (Hu et al., 2023). The key goal is to forecast future actions based on recent experiences. Chen et al. (2021) applied a Transformer to a context-conditioned, model-free policy, while Janner et al. (2021) emphasized the sequence model's ability to predict states, actions, and rewards with beam search. Later studies fine-tune the Transformer for online environments (Zheng et al., 2022), and StAR-representations were developed for enhanced long-term modeling (Shang et al., 2022). Our proposed method improves decision transformer-based RL models by integrating return-to-go into the attention mechanism, enhancing data processing efficiency and offering a unique improvement over existing sequence modeling techniques.

Multimodal Data and Planning   Exploring the intersection of language with vision and text in RL is a recent trend, with studies increasingly incorporating NLP into the RL domain (Sun et al., 2022; Liu et al., 2022b; Wang et al., 2023a; Jin et al., 2024). This research area treats PLMs as offline RL agents and aims to separate high-level planning from modalities.

LPM for RL enables better reward descriptions, environment exploration, and agent communication, resulting in improved RL outcomes (Huang et al., 2022; Reed et al., 2022; ichter et al., 2023; Liu et al., 2022a). Findings suggest a close link between language, local sequence relationships, and high-level planning across modalities (Reid et al., 2022; Tsimpoukelli et al., 2021; Zhang et al., 2022). In this context, Liu et al. (2022a) explore a shared transformer architecture for vision and text games, bringing in the strength of NLP to the
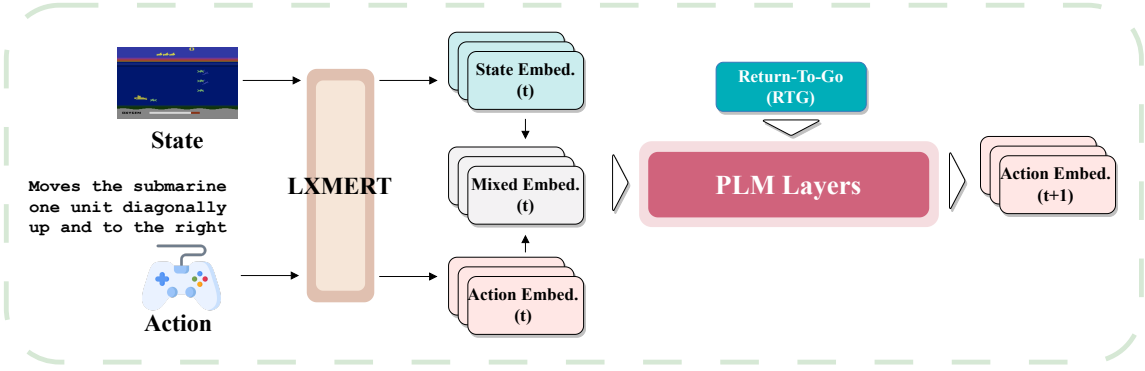
Figure 1: Architecture diagram of the proposed MORE-3S approach. The Multimodal Encoder component combines the action (text) and state (image) inputs using the LXMERT model. "Embed." denotes the embedding process. Autoregressive modeling of trajectories captures the system's dynamics by modeling trajectories as a sequence of tuples. LPMs predict subsequent actions based on the encoded sequence O_t, which corresponds to the 'Mixed Embed.' section in the diagram.

visual RL landscape.

Our work distinguishes itself in a crucial aspect: the implicit modality alignment. While previous studies have explicitly used pre-trained models like CLIP (Radford et al., 2021; Lee et al., 2022) for vision and language tasks in RL, our approach leverages implicit modality alignment by coupling latent state representations derived from images with their corresponding textual descriptions in the action space. This enables the model to achieve a more nuanced understanding of states and actions, ultimately enhancing RL performance. Unlike conventional approaches, which align modalities explicitly and in isolation (Jiang et al., 2020; Zeng et al., 2020), our method inherently aligns these modalities through the training process. This novel implicit alignment allows for a more flexible integration of vision and text, addressing the complexity of real-world RL applications more effectively.

3. Preliminary

Trajectory Rollouts in Offline RL In Offline RL, a trajectory is characterized as a sequence of state-action-reward triplets, represented as $D = \{(s_t, a_t, r_t)\}_{t=0}^T$. Here, s_t indicates the state, a_t the action executed, and r_t the reward received at time step t. The dataset D comprises trajectories up to the time horizon T, with each trajectory uniquely identified by index i.

Sequence Representation in RL In sequence modeling for RL, a sequence τ of experiences is used to predict the next actions. This sequence can be represented as follows:

$$\tau = (r_1, s_1, a_1, r_2, s_2, a_2, \ldots, r_N, s_N, a_N) \quad (1)$$

This representation includes state-action-reward triplets and is suitable for offline RL and imitation

learning scenarios, where the model can be trained in a supervised manner. The goal of sequence modeling in RL is to predict prospective actions based on historical experiences. This can be formulated (Janner et al., 2021):

$$Pr(\hat{a}_t) = p(a_t | s_{1:t}, a_{1:t-1}, r_{1:t-1}) \quad (2)$$

Transformers in RL Transformers are neural network architectures that are highly successful in natural language processing tasks. Their ability to capture long-range dependencies in data makes them particularly suitable for sequence modeling in RL. In this context, Transformers can be used to model the relationships between state-action-reward triplets in trajectories and make predictions about future actions and rewards.

Multimodal Context in RL Incorporating multimodal data such as text and vision can enhance the capabilities of RL agents. This can be particularly useful in complex environments where an agent needs to make decisions based on multiple sources of information. LPM can be adapted to function as Offline RL agents in a multimodal context, leveraging their capacity for high-level planning and local sequence modeling.

4. Architecture and Training

In this section, we outline the MORE-3S approach. MORE-3S synergizes multimodal and pre-trained sequence models to enhance offline RL. Figure 1 illustrates the overall process, including state and action interpretation, as well as action prediction.

4.1. Model Architecture

The model structure in this proposed Multimodal-based Offline Reinforcement Learning approach

involves the integration of multimodal models and pre-trained sequence models into a Reinforcement Learning framework, where the multimodal model employs pre-training parameters to interpret state information from image data and action information from text, and the output of this multimodal model serves as the input for the LPM to predict subsequent actions.

Multimodal Encoder In the initial segment of MORE-3S, we delve into the two primary inputs at time t: the action and the state. These inputs are subsequently converted into their respective forms, with the action being transformed into text and the state into an image. To encode these inputs, we utilize the LXMERT (Tan and Bansal, 2019) model, a robust multimodal transformer model pre-trained on a substantial corpus of image-text pairs. However, Nota Bene that MORE-3S's strength lies in the general approach of using a multimodal transformer model instead of the specific LXMERT. Upon encoding, the actions and states are amalgamated into a single sequence O_t, Specifically,

$$O_t = \text{Concat}(E(A_t), E(S_t)) \qquad (3)$$

where $E(A_t)$ and $E(S_t)$ represent the encoded forms of the action and the state at time t, respectively.serving as the input for the subsequent component of our model.

Autoregressive Modeling of Trajectories In the subsequent section of our study, we adopt the methodology delineated by Chen et al. (2021), wherein an autoregressive approach is employed to model trajectories. This necessitates the depiction of these trajectories as a sequence of tuples, formally represented as:

$$t = (\hat{R}_1, O_1, \hat{R}_2, O_2, ..., \hat{R}_N, O_N) \qquad (4)$$

Each tuple in this sequence encapsulates the returns-to-go ($\hat{R}_i$) and the decoupled information pertaining to the state and action (O_i) at a specific timestep i, given a total of N timesteps.

This modeling strategy draws parallels with a sequence x, as explicated in the equation proposed by Bengio et al. (2000):

$$P(\mathbf{x}) = \prod_{i=1}^{N} p\left(\mathbf{x}_i \mid \mathbf{x}_{i-1}, \mathbf{x}_{i-2}, \ldots, \mathbf{x}_1\right) \qquad (5)$$

This equation embodies the fundamental principles of a self-regressive language model. The returns-to-go, $\hat{R}_i$, is mathematically represented as $\hat{R}_i = \sum_{t=i}^{N} r_t$. By incorporating the processed information from the state and action (O_N) and the return-to-go (R_N) into our model, we effectively capture the dynamics of the system under investigation. This approach allows us to construct a comprehensive model that is capable of accurately representing the system's behavior over time.

4.2. Memory Mechanism

To improve our model's proficiency in managing lengthy sequences, we integrated a memory mechanism inspired by Wu et al. (2022) into the GPT-style attention architecture. This memory approach enhances the model's ability to address long-term dependencies, as it facilitates storing and accessing information from prior stages of a sequence.

Incorporating the return-to-go quantity into the original keys K and values V of the transformer architecture has proven to be more effective than directly inputting it during the model's position embedding stage,see Figure 2. This process leverages the attention weights at each time step t to maintain a "memory" of earlier inputs, thus improving the model's capability to process tasks requiring long-term dependency handling.

The memory mechanism is practically implemented by extending the keys $\bar{K}$ and values $\bar{V}$ at the current step t to include $\bar{K}(t')$ and $\bar{V}(t')$ from previous steps t', yielding:

$$\bar{V}^t = \left[\text{sg}\left(\bar{V}^{(t-M)}\right), \cdots, \text{sg}\left(\bar{V}^{(t-1)}\right), \bar{V}^t\right] \qquad (6)$$

$$\bar{K}^t = \left[\text{sg}\left(\bar{K}^{(t-M)}\right), \cdots, \text{sg}\left(\bar{K}^{(t-1)}\right), \bar{K}^t\right] \qquad (7)$$

This formulation allows the query to attend information not only from the current time step t, but also from up to M preceding steps, thereby offering an extended view(see Fiture 2). The increase in training and inference costs is primarily due to memory caching and the extended attention layer computation, but remains more efficient than traditional scaling methods, making our model practically suitable for diverse tasks.
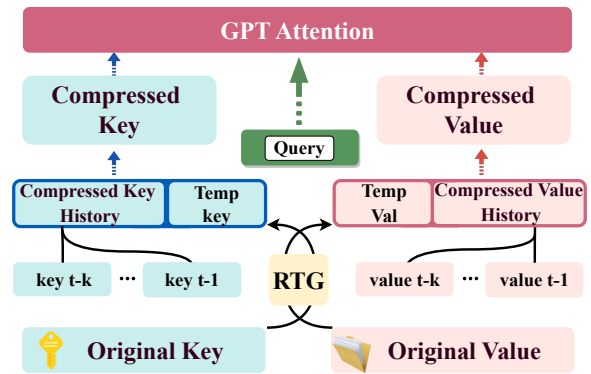

Figure 2: Schematic Representation of the Integration of Return-to-Go (**RTG**) Quantity and Memory Mechanism in GPT-style Attention Architecture.

4.3. Training Details

Object-Level State Embeddings Our approach involves utilizing position features and region-of-interest (**RoI**) features obtained from image-based

states through a Fast R-CNN (Girshick, 2015) network as input to our model. This approach differs from the conventional usage of feature maps generated by convolutional neural networks. Instead, we adopt the object feature-based embeddings proposed by Anderson et al. (2018). By integrating object and position features, our method facilitates a more comprehensive comprehension of the image states, resulting in enhanced performance.

Semantic Embedding of Actions   To enhance the semantic comprehension of actions in the multimodal model LXMERT, we represent actions using both discrete and continuous textual descriptions. These representations facilitate a clearer understanding of actions, benefiting the model's comprehension and facilitating human interpretability.

As an exemplar, consider the actions within the game Breakout from the Atari 2600 suite:

- **Action 0**: No action is taken, allowing the game to continue unchanged.

- **Action 1**: Launches a ball towards the bricks, aiming to break them.

- **Action 2**: Shifts the paddle to the right, intercepting the ball to prevent it from falling.

- **Action 3**: Shifts the paddle to the left, intercepting the ball to prevent it from falling.

For continuous environments based on the Mujoco framework, such as the HalfCheetah simulation, we provide textual prompts mapping specific dimensions of the action vector to torques or forces applied to particular joints. For instance, in the HalfCheetah environment, a prompt might state: "In the current time step, the Half-Cheetah's six joints received the following torques or forces: First hind leg joint: a1, Second hind leg joint: a2, etc."

Each of these textual descriptions is generated from specific prompts, translating the numerical values in each dimension to the corresponding actions taken in the environment.

4.4.   Procedure of Training

In the training phase, the primary objective is to leverage the multimodal encoder in conjunction with the sequence model to accurately predict the subsequent timestep's output, O_{i+1}. Importantly, the multimodal encoder is not subject to training; its parameters are fixed during this phase. The multimodal encoder processes the state and action data, represented as image and text respectively, at each timestep i, yielding an encoded output, O_i. This output, in tandem with the corresponding returns-to-go, denoted as $hatR_i$, serves as the input for the GPT-style sequence model.

Incorporating $\hat{R}_i$ into its attention mechanism, the sequence model can adjust its predictions contingent on the anticipated future returns. The model's primary task is to forecast the subsequent timestep's output, O_{i+1}, using the mean squared error (**MSE**) loss function as our objective. The MSE loss function is a natural choice for this supervised learning task, as it minimizes the squared differences between predicted and actual values. The predicted output distribution, O_{i+1}^{pred}, is compared with the actual output distribution, O_{i+1}, to calculate the prediction error as follows:

$$\mathcal{L} = \frac{1}{N}\sum_{i=1}^{N}(O_{i+1} - O_{i+1}^{pred})^2 \qquad (8)$$

By repeatedly minimizing the loss function $\mathcal{L}$, it becomes possible to incrementally enhance the model's predictions.

5.   Experiments

This section outlines our experimental setup, the benchmarks we employed, our evaluation metrics, and the implementation details of MORE-3S.

5.1.   Experimental Setup

Our experimental setup is constructed to evaluate our model in diverse environments, with a broad set of tasks that require different skill sets. We employ Atari 2600 games (Bellemare et al., 2013) and OpenAI Gym environments (Brockman et al., 2016), which encompass a wide range of complexity levels and task varieties, offering a comprehensive benchmark suite for our evaluation.

MORE-3S utilizes distinct pretraining models - LXMERT and a GPT-inspired model - for the optimization of RL training efficiency. For LXMERT, the training spans an extensive range of datasets, hence fostering a comprehensive understanding of both visual and linguistic elements. We exclusively utilize the outputs of the Object Relationship Encoder and Language Encoder which are subsequently concatenated and integrated

For the GPT-inspired model, we use only the first six layers of the 12-layer GPT-2 model, allowing for a direct comparison with the DT and capitalizing on its strengths in sequence modeling for RL tasks (Chen et al., 2021). Pre-training of MORE-3S employs a model with 128 dimensions in its latent space, a solitary attention head, and six layers. The training, conducted with a learning rate of 3e-4 and a batch size comprising 65,536 tokens in each training batch, incorporates a warm-up schedule in the initial 10,000 steps. We utilize the Byte-Pair Encoding (Sennrich et al., 2016), also used in GPT-2 (Radford et al., 2019) and follow all hyperparameters given in (Chen et al., 2021).

5.2. Baselines

To augment our assessment of LPM, we conduct a comparative analysis against a suite of widely acclaimed offline RL algorithms. This collection of algorithms encompasses the DT, Conservative Q-Learning (CQL), Twin Delayed Deep Deterministic policy gradient enhanced with Behavior Cloning (TD3+BC), Behavior Regularized Actor Critic (BRAC), and Advantage Weighted Regression (AWR) baselines.

5.3. Atari

Our model is applied to the renowned Atari benchmark, a complex dataset developed by Bellemare et al. (2013), to evaluate its performance in managing high-dimensional visual inputs and intricate credit assignments. Four games: Breakout, Pong, Qbert, and Seaquest from the Atari suite are used for the testing, following the evaluation design in (Chen et al., 2021; Agarwal et al., 2020).

An offline dataset is constructed for the current study, consisting of approximately 1% (or 500k transition steps) of the replay buffer dataset used in (Agarwal et al., 2020). Benchmark metrics are derived from various sources, including the Conservative Q-Learning (CQL), Random Ensemble Mixture (REM), and Quantile Regression DQN (QR-DQN) numbers from Kumar et al. (2020); Agarwal et al. (2020). Additionally, we referenced imitation learning algorithms such as DT, StARformer, and BC.

Performance comparison of MORE-3S, with CQL, QR-DQN, REM, BC, DT, and StAR is detailed in Table 1. Remarkably, MORE-3S surpass all other models in Breakout and Qbert and achieved competitive results in Pong and Seaquest. This indicates the effectiveness of MORE-3S, which combines multimodal data during training and uses multimodal models to process visual and textual information. The model, initialized with pre-trained GPT-style parameters, shows significant improvement in Qbert and Breakout, emphasizing the value of incorporating multimodal information and pre-trained architectures in sequential reinforcement learning. This establishes the robustness of MORE-3S and underscores the successful implementation of pretraining and multimodal data utilization.

5.4. OpenAI Gym

Our research employs the OpenAI Gym, a reputable platform that facilitates the development and comparison of RL algorithms across diverse environments. We evaluate three standard environments (HalfCheetah, Hopper, Walker), selected for their challenging control and decision-making characteristics. These are assessed using three dataset configurations, medium, medium-replay, and medium-expert, in line with the D4RL benchmark proposed by Fu et al. (2021). This benchmark standardizes scores for fair comparisons, where a score of 100 signifies an expert policy.

We evaluate our proposed model, MORE-3S, against several state-of-the-art algorithms to benchmark its performance. This includes model-free methodologies (CQL, BEAR, BRAC-v, AWR), a model-based approach (MBOP), and imitation learning algorithms (DT, StARformer). Performance metrics are derived from original or D4RL papers, and additional methods supplement our comprehensive evaluation. The experimental protocol comprise training models for 100k timesteps and conducting evaluations every 5k timesteps over 10 episodes. Using the formula $100 \times (score - random\ score)/(expert\ score - random\ score)$, normalized return scores are calculated to gauge MORE-3S's efficiency in continuous control tasks and its performance against leading algorithms.

Table 2 illustrates MORE-3S's efficacy across D4RL's various environments. It demonstrates MORE-3S's superior performance on most tasks, with comparable results to the state-of-the-art algorithm on others. MORE-3S's unique feature is its integration of multimodal information processing mechanisms, which capacitates it to manage diverse inputs and structures. Pre-training a language model for reinforcement learning, reminiscent of GPT-style models (Radford et al., 2019), augments MORE-3S's capability. By leveraging the language models' knowledge via multimodal processing, MORE-3S effectively captures fine-grained spatial information and adapts to different environments. The results confirm MORE-3S's competitiveness and, in many cases, superiority to other state-of-the-art algorithms.

6. Discussion

Through an exhaustive ablation study, we systematically dissect our computational model to understand the individual contributions of its various components. Our in-depth analysis evaluate the effects of implementing a condition-based return-to-go strategy, integrating pretrained GPT-style parameters, and applying LXMERT's pretrained parameters for state and action data preprocessing. We also examined the role of employing long-term attention mechanisms and assessed the implications of both context length and model size on overall performance. Additionally, we conduct ablation studies across multiple games on the use of synonyms and contextual phrasing to explore the impact of varying action prompts. The experiment results indicate that varying action prompts can have little impact on MORE-3S's performance, which verifies MORE-3S's robustness.

Game	MORE-3S	CQL	QR-DQN	REM	BC	DT	StAR
Breakout	300.5 ± 70.2	211.1	17.1	8.9	138.9 ± 61.7	267.5 ± 97.5	**436.1 ± 63.6**
Qbert	60.8 ± 10.2	**104.2**	0	0	17.3 ± 14.7	15.4 ± 11.4	51.2 ± 11.5
Pong	**118.7 ± 9.6**	111.9	18	0.5	85.2 ± 20.0	106.1 ± 8.1	110.8 ± 60.3
Seaquest	**3.1 ± 0.3**	1.7	0.4	0.7	2.1 ± 0.3	2.5 ± 0.4	1.7 ± 0.3

Table 1: Performance comparison of MORE-3S with models such as CQL, QR-DQN, REM, BC, DT, and StAR in four Atari games, using three seeds. MORE-3S's use of multimodal information and GPT-style pre-training led to high scores, highlighting its strong potential in gaming. Best mean scores are in bold and second-best mean scores are in gray bold.

Dataset	Environment	MORE-3S	CQL	BEAR	BRAC-v	AWR	MBOP	BC	DT	StAR
Medium Expert	HalfCheetah	93.5 ± 1.0	62.4	53.4	41.9	52.7	**105.9**	59.9	86.8 ± 1.3	93.7 ± 0.1
	Hopper	**112.2 ± 0.8**	111.0	96.3	0.8	27.1	55.1	79.6	107.6 ± 1.8	111.1 ± 0.2
	Walker	**111.3 ± 0.1**	98.7	40.1	81.6	53.8	70.2	36.6	108.1 ± 0.2	109.0 ± 0.1
Medium	HalfCheetah	42.8 ± 0.1	44.4	41.7	**46.3**	37.4	44.6	43.1	42.6 ± 0.1	42.9 ± 0.1
	Hopper	**76.4 ± 2.5**	58.0	52.1	31.1	35.9	48.8	63.9	67.6 ± 1.0	59.5 ± 4.2
	Walker	**81.5 ± 0.1**	79.2	59.1	81.1	17.4	41.0	77.3	74.0 ± 1.4	73.8 ± 3.5
Medium Replay	HalfCheetah	36.5 ± 0.4	46.2	38.6	**47.7**	40.3	42.3	4.3	36.6 ± 0.8	36.8 ± 3.3
	Hopper	**85.1 ± 2.7**	48.6	33.7	0.6	28.4	12.4	27.6	82.7 ± 7.0	29.2 ± 4.3
	Walker	**76.9 ± 1.5**	26.7	19.2	0.9	15.5	9.7	36.9	66.6 ± 3.0	39.8 ± 5.1
Average		**79.6**	63.9	48.2	36.9	34.3	47.8	46.4	74.7	66.2

Table 2: Results from the D4RL dataset reveal the average and variance across three seeds. Best mean scores are highlighted in bold. The second-best scores are highlighted in gray bold.

To secure a robust evaluation, we execute a series of ablation experiments across computational environments. Our findings suggest that extending context length and model size beyond certain limits does not significantly enhance performance.

Condition-Based RTG We examine two distinct strategies for integrating the return-to-go metric into our model: condition-based and linear layer integration. In the condition-based methodology, each temporal step t processes the return-to-go through a linear layer, which is subsequently integrated into the transformer's native keys K and values V. The linear layer approach, conversely, incorporates the return-to-go directly into the initial position embedding stage of the model.

The ablation study demonstrates a modest performance benefit of condition-based integration compared to the linear layer approach, as shown in Table 3. This modest advantage indicates that, while the condition-based method is superior in managing long-term dependencies, it does not produce markedly improved results. Consequently, the selection between the two methods should be based on the specific needs of the given RL task.

Randomizing Pretrained Parameters Our ablation study examines the effect of randomizing the pretrained parameters of GPT-style and LXMERT models on performance, as depicted in Figure 3. When the pretrained parameters of LXMERT are frozen, the model demonstrates enhanced perfor-

Model	Avg. Reward
MORE-3S (Condition-Based)	66.9
MORE-3S (Linear layer integration)	65.4

Table 3: Comparison of condition-based and linear-layer integration of return-to-go.

mance in processing multimodal data. This leads to improved interpretation of both state and action information, resulting in better performance across all evaluated environments compared to using randomly initialized LXMERT parameters. Contrarily, when the GPT-style's pretrained parameters are absent, the model's initial performance is subpar, reminiscent of the DT(Chen et al., 2021) model. However, with sustained training, GPT-style's performance, although not superior, tends to converge towards that of our fully pretrained model. This observation implies that even though pretraining GPT-style's parameters provides an initial performance boost, the model possesses the capability to effectively learn from scratch over time. This finding further supports the robustness of our proposed method of integrating pretrained sequence models into sequential RL in diverse training conditions.

Long-Term Attention We show the results of ablation studies comparing the performance of MORE-3S without long-term attention with full-fledged MORE-3S in Table 4 to ascertain the effi-

cacy of our proposed memory mechanism and the long-term attention module.

Our analysis illustrates that the model without a long-term mechanism often underperforms MORE-3S across numerous Atari games, affirming the memory mechanism and long-term attention's role in managing sequential data for RL tasks. Interestingly, MORE-3S without long-term components performs better on Seaquest, suggesting possible instances where long-term attention may not be vital. This observation necessitates further research.

Model	MORE-3S	MORE-3S$_{\text{w/o Long-Term}}$
Breakout	**300.5 ± 70.2**	275.7 ± 66.9
Qbert	**60.8 ± 10.2**	55.3 ± 7.3
Pong	**118.7 ± 9.6**	105.2 ± 11.1
Seaquest	3.1 ± 0.3	**3.3 ± 0.3**

Table 4: The influence of the long-term attention component across different RL tasks.

Varying Action Prompts in Game Environments
In our quest to discern the model's aptitude for linguistic comprehension, we venture into experiments with varied action prompt sets for each game, as showcased in Table 5. Each game is subjected to three distinct action prompt categories:

- **Original**: Action prompts as originally defined.

- **Synonyms**: Prompts incorporating synonymous replacements for pivotal action terms.

- **Contextual**: Prompts with deeper context or phrased as interrogative statements.

Taking the game Breakout as a reference, the prompt sets varied as:

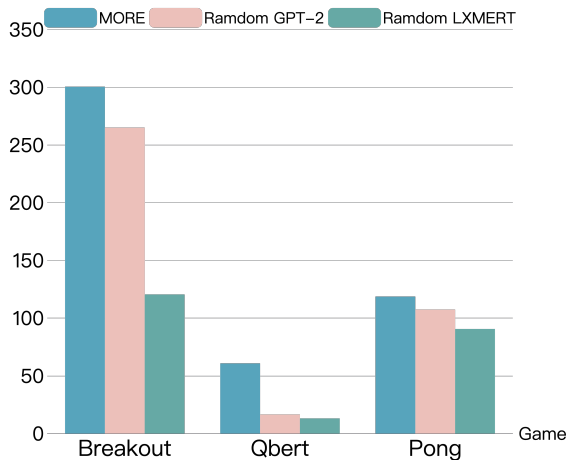- Original: "Launches a ball towards the bricks, aiming to break them."

Figure 3: Experiment on randomizing model weights versus finetuning them on OpenAI Gym.

- Synonyms: "Fires a sphere at the blocks with the intent to shatter them."

- Contextual: "What would happen if a ball were propelled towards the bricks with the goal of breaking them?"

Game	Action Prompt Set	Reward
Breakout	Original	310.3 ± 68.1
	Synonyms	295.4 ± 72.3
	Contextual	305.2 ± 69.5
Pong	Original	120.1 ± 8.5
	Synonyms	117.9 ± 10.2
	Contextual	119.5 ± 9.8
Qbert	Original	58.5 ± 12.3
	Synonyms	63.2 ± 11.1
	Contextual	59.7 ± 10.5
Seaquest	Original	3.0 ± 0.4
	Synonyms	2.9 ± 0.5
	Contextual	3.2 ± 0.2

Table 5: Results of Experiments with Different Action Prompt Sets

Upon closely examining the results, it is evident that there is minimal variance in performance across different prompt sets. Whether employing standard terminology, synonyms, or more context-rich queries, the model's performance is consistent. This uniformity underscores the model's linguistic proficiency and its adaptability to subtle linguistic variations in a gamified environment. These findings reinforce the reliability and robustness of our experimental method and suggest the model's superior linguistic interpretation in diverse scenarios.

7. Conclusion

In conclusion, this research underscores the symbiotic relationship between RL and natural NLP, elucidating the tangible advantages conferred upon RL tasks when augmented with NLP models. Our pioneering MORE-3S method seamlessly amalgamates multimodal models with pre-trained sequence models in an RL milieu, delivering a compelling empirical corroboration for our theoretical postulations through demonstrably superior RL training outcomes. Instead of merely capitalizing on historical state-action-reward sequences, our methodology elevates these dynamics by anchoring the actions in textually-described formats and the states in visual representations, thereby bolstering the model's precision and capability in interacting with its environment. A noteworthy innovation we bring forth is the integration of the return-to-go parameter within the attention framework of decision transformer-centric RL models, further refining

their performance. This scholarly endeavor illuminates potential avenues for an in-depth examination of the confluence between language models and reinforcement learning and heralds exciting prospects for the continued evolution of both offline RL and NLP domains.

Bibliographical References

Rishabh Agarwal, Dale Schuurmans, and Mohammad Norouzi. 2020. An optimistic perspective on offline reinforcement learning. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 104–114. PMLR.

Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2018. Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

G. E. M. Anscombe. 1985. Wittgenstein on rules and private language. *Ethics*, 95(2):342–352.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. 2015. Vqa: Visual question answering. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

Marc G. Bellemare, Yavar Naddaf, Joel Veness, and Michael Bowling. 2013. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, 47.

Yoshua Bengio, Réjean Ducharme, and Pascal Vincent. 2000. A neural probabilistic language model. In *Advances in Neural Information Processing Systems*, volume 13. MIT Press.

Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. 2016. Openai gym.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner,

Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Lili Chen, Kevin Lu, Aravind Rajeswaran, Kimin Lee, Aditya Grover, Misha Laskin, Pieter Abbeel, Aravind Srinivas, and Igor Mordatch. 2021. Decision transformer: Reinforcement learning via sequence modeling. In *Advances in Neural Information Processing Systems*, volume 34, pages 15084–15097. Curran Associates, Inc.

André Correia and Luís A. Alexandre. 2022. Hierarchical decision transformer.

Linxi Fan, Guanzhi Wang, Yunfan Jiang, Ajay Mandlekar, Yuncong Yang, Haoyi Zhu, Andrew Tang, De-An Huang, Yuke Zhu, and Anima Anandkumar. 2022. Minedojo: Building open-ended embodied agents with internet-scale knowledge. In *Advances in Neural Information Processing Systems*, volume 35, pages 18343–18362. Curran Associates, Inc.

Justin Fu, Aviral Kumar, Ofir Nachum, George Tucker, and Sergey Levine. 2021. D4rl: Datasets for deep data-driven reinforcement learning.

Scott Fujimoto, Edoardo Conti, Mohammad Ghavamzadeh, and Joelle Pineau. 2019. Benchmarking batch deep reinforcement learning algorithms.

Scott Fujimoto and Shixiang (Shane) Gu. 2021. A minimalist approach to offline reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 34, pages 20132–20145. Curran Associates, Inc.

Ross Girshick. 2015. Fast r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks.

Shengchao Hu, Li Shen, Ya Zhang, Yixin Chen, and Dacheng Tao. 2023. On transforming reinforcement learning by transformer: The development trajectory.

Wenlong Huang, Pieter Abbeel, Deepak Pathak, and Igor Mordatch. 2022. Language models as zero-shot planners: Extracting actionable knowledge for embodied agents. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 9118–9147. PMLR.

brian ichter, Anthony Brohan, Yevgen Chebotar, Chelsea Finn, Karol Hausman, Alexander Herzog, Daniel Ho, Julian Ibarz, Alex Irpan, Eric Jang, Ryan Julian, Dmitry Kalashnikov, Sergey Levine, Yao Lu, Carolina Parada, Kanishka Rao, Pierre Sermanet, Alexander T Toshev, Vincent Vanhoucke, Fei Xia, Ted Xiao, Peng Xu, Mengyuan Yan, Noah Brown, Michael Ahn, Omar Cortes, Nicolas Sievers, Clayton Tan, Sichun Xu, Diego Reyes, Jarek Rettinghouse, Jornell Quiambao, Peter Pastor, Linda Luu, Kuang-Huei Lee, Yuheng Kuang, Sally Jesmonth, Nikhil J. Joshi, Kyle Jeffrey, Rosario Jauregui Ruano, Jasmine Hsu, Keerthana Gopalakrishnan, Byron David, Andy Zeng, and Chuyuan Kelly Fu. 2023. Do as i can, not as i say: Grounding language in robotic affordances. In *Proceedings of The 6th Conference on Robot Learning*, volume 205 of *Proceedings of Machine Learning Research*, pages 287–318. PMLR.

Michael Janner, Qiyang Li, and Sergey Levine. 2021. Offline reinforcement learning as one big sequence modeling problem. In *Advances in Neural Information Processing Systems*, volume 34, pages 1273–1286. Curran Associates, Inc.

Xiang Jiang, Qicheng Lao, Stan Matwin, and Mohammad Havaei. 2020. Implicit class-conditioned domain alignment for unsupervised domain adaptation. In *International Conference on Machine Learning*, pages 4816–4827. PMLR.

Yonggang Jin, Ge Zhang, Hao Zhao, Tianyu Zheng, Jiawei Guo, Liuyu Xiang, Shawn Yue, Stephen W. Huang, Wenhu Chen, Zhaofeng He, and Jie Fu. 2024. Read to play (r2-play): Decision transformer with multimodal game instruction. *CoRR*, abs/2402.04154.

Rahul Kidambi, Aravind Rajeswaran, Praneeth Netrapalli, and Thorsten Joachims. 2020. Morel: Model-based offline reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 33, pages 21810–21823. Curran Associates, Inc.

Ilya Kostrikov, Rob Fergus, Jonathan Tompson, and Ofir Nachum. 2021. Offline reinforcement learning with fisher divergence critic regularization. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 5774–5783. PMLR.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Li Fei-Fei. 2017. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123(1):32–73.

Taku Kudo and John Richardson. 2018. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing.

Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. 2020. Conservative q-learning for offline reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 33, pages 1179–1191. Curran Associates, Inc.

Kuang-Huei Lee, Ofir Nachum, Mengjiao (Sherry) Yang, Lisa Lee, Daniel Freeman, Sergio Guadarrama, Ian Fischer, Winnie Xu, Eric Jang, Henryk Michalewski, and Igor Mordatch. 2022. Multi-game decision transformers. In *Advances in Neural Information Processing Systems*, volume 35, pages 27921–27936. Curran Associates, Inc.

Sergey Levine, Aviral Kumar, George Tucker, and Justin Fu. 2020. Offline reinforcement learning: Tutorial, review, and perspectives on open problems.

Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. 2015. Microsoft coco: Common objects in context.

Ruibo Liu, Jason Wei, Shixiang Shane Gu, Te-Yen Wu, Soroush Vosoughi, Claire Cui, Denny Zhou, and Andrew M. Dai. 2022a. Mind's eye: Grounded language model reasoning through simulation.

Ruibo Liu, Ge Zhang, Xinyu Feng, and Soroush Vosoughi. 2022b. Aligning generative language models with human values. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 241–252, Seattle, United States. Association for Computational Linguistics.

Kevin Lu, Aditya Grover, Pieter Abbeel, and Igor Mordatch. 2021. Pretrained transformers as universal computation engines. *arXiv preprint arXiv:2103.05247*, 1.

OpenAI. 2023. Gpt-4 technical report.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language

models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

Christos H Papadimitriou and John N Tsitsiklis. 1987. The complexity of markov decision processes. *Mathematics of operations research*, 12(3):441–450.

Xue Bin Peng, Aviral Kumar, Grace Zhang, and Sergey Levine. 2019. Advantage-weighted regression: Simple and scalable off-policy reinforcement learning.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training. Technical report, OpenAI.

Alec Radford, Jeff Wu, Rewon Child, D. Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Scott Reed, Konrad Zolna, Emilio Parisotto, Sergio Gomez Colmenarejo, Alexander Novikov, Gabriel Barth-Maron, Mai Gimenez, Yury Sulsky, Jackie Kay, Jost Tobias Springenberg, Tom Eccles, Jake Bruce, Ali Razavi, Ashley Edwards, Nicolas Heess, Yutian Chen, Raia Hadsell, Oriol Vinyals, Mahyar Bordbar, and Nando de Freitas. 2022. A generalist agent.

Machel Reid, Yutaro Yamada, and Shixiang Shane Gu. 2022. Can wikipedia help offline reinforcement learning?

Martin Riedmiller. 2005. Neural fitted q iteration – first experiences with a data efficient neural reinforcement learning method. In *Machine Learning: ECML 2005*, volume 3720, pages 317–328, Berlin, Heidelberg. Springer Berlin Heidelberg.

Thomas Schmied, Markus Hofmarcher, Fabian Paischer, Razvan Pascanu, and Sepp Hochreiter. 2023. Learning to modulate pre-trained models in rl.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units.

Jinghuan Shang, Kumara Kahatapitiya, Xiang Li, and Michael S. Ryoo. 2022. Starformer: Transformer with state-action-reward representations for visual reinforcement learning.

Rupesh Kumar Srivastava, Pranav Shyam, Filipe Mutz, Wojciech Jaśkowski, and Jürgen Schmidhuber. 2021. Training agents using upside-down reinforcement learning.

Yuxiang Sun, Bo Yuan, Qi Xiang, Jiawei Zhou, Jiahui Yu, Di Dai, and Xianzhong Zhou. 2022. Intelligent decision-making and human language communication based on deep reinforcement learning in a wargame environment. *IEEE Transactions on Human-Machine Systems*, 53(1):201–214.

Hao Tan and Mohit Bansal. 2019. Lxmert: Learning cross-modality encoder representations from transformers.

Maria Tsimpoukelli, Jacob L Menick, Serkan Cabi, SM Eslami, Oriol Vinyals, and Felix Hill. 2021. Multimodal few-shot learning with frozen language models. *Advances in Neural Information Processing Systems*, 34:200–212.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Kerong Wang, Hanye Zhao, Xufang Luo, Kan Ren, Weinan Zhang, and Dongsheng Li. 2022. Bootstrapped transformer for offline reinforcement learning. *Advances in Neural Information Processing Systems*, 35:34748–34761.

Zekun Wang, Ge Zhang, Kexin Yang, Ning Shi, Wangchunshu Zhou, Shaochun Hao, Guangzheng Xiong, Yizhi Li, Mong Yuan Sim, Xiuying Chen, Qingqing Zhu, Zhenzhu Yang, Adam Nik, Qi Liu, Chenghua Lin, Shi Wang, Ruibo Liu, Wenhu Chen, Ke Xu, Dayiheng Liu, Yike Guo, and Jie Fu. 2023a. Interactive natural language processing.

Zihao Wang, Shaofei Cai, Anji Liu, Xiaojian Ma, and Yitao Liang. 2023b. Describe, explain, plan and select: Interactive planning with large language models enables open-world multi-task agents.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837.

Ludwig Wittgenstein. 1953. *Philosophical Investigations*. Wiley-Blackwell.

Maciej Wolczyk, Michał Zając, Razvan Pascanu, Łukasz Kuciński, and Piotr Miłoś. 2022. Disentangling transfer in continual reinforcement learning. *Advances in Neural Information Processing Systems*, 35:6304–6317.

Chao-Yuan Wu, Yanghao Li, Karttikeya Mangalam, Haoqi Fan, Bo Xiong, Jitendra Malik, and Christoph Feichtenhofer. 2022. Memvit: Memory-augmented multiscale vision transformer for efficient long-term video recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13587–13597.

Yifan Wu, George Tucker, and Ofir Nachum. 2019. Behavior regularized offline reinforcement learning.

Yue Wu, Shuangfei Zhai, Nitish Srivastava, Joshua Susskind, Jian Zhang, Ruslan Salakhutdinov, and Hanlin Goh. 2021. Uncertainty weighted actor-critic for offline reinforcement learning.

Taku Yamagata, Ahmed Khalil, and Raul Santos-Rodriguez. 2023. Q-learning decision transformer: Leveraging dynamic programming for conditional sequence modelling in offline rl. In *International Conference on Machine Learning*, pages 38989–39007. PMLR.

Yiqin Yang, Xiaoteng Ma, Chenghao Li, Zewu Zheng, Qiyuan Zhang, Gao Huang, Jun Yang, and Qianchuan Zhao. 2021. Believe what you see: Implicit constraint approach for offline multi-agent reinforcement learning. *Advances in Neural Information Processing Systems*, 34:10299–10312.

Tianhe Yu, Garrett Thomas, Lantao Yu, Stefano Ermon, James Y Zou, Sergey Levine, Chelsea Finn, and Tengyu Ma. 2020. Mopo: Model-based offline policy optimization. *Advances in Neural Information Processing Systems*, 33:14129–14142.

Zhen Zeng, Jianzong Wang, Ning Cheng, Tian Xia, and Jing Xiao. 2020. Aligntts: Efficient feedforward text-to-speech system without explicit alignment. In *ICASSP 2020-2020 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 6714–6718. IEEE.

Qin Zhang, Linrui Zhang, Haoran Xu, Li Shen, Bowen Wang, Yongzhe Chang, Xueqian Wang, Bo Yuan, and Dacheng Tao. 2023. Saformer: A conditional sequence modeling approach to offline safe reinforcement learning.

Ziqi Zhang, Yile Wang, Yue Zhang, and Donglin Wang. 2022. Can offline reinforcement learning help natural language understanding?

Qinqing Zheng, Amy Zhang, and Aditya Grover. 2022. Online decision transformer. In *international conference on machine learning*, pages 27042–27059. PMLR.