

Latent Abstraction for Optimization and Generation

PI: Ying Nian Wu, UCLA Department of Statistics and Data Science

Overview:

This project develops a unified framework for incorporating explicit latent abstractions into modern artificial intelligence systems. While current deep learning approaches have achieved remarkable success through memorization and interpolation, they often lack explicit abstraction capabilities that are crucial for human-like generalization and reasoning. Drawing inspiration from how scientific theories use latent concepts (like force and mass in physics) to enable understanding and generalization, we propose to enhance AI systems with learned abstract representations that guide generation and optimization. Our framework introduces different manifestations of latent abstraction - "designs" for optimization problems, "plans" for sequential decision making, "thoughts" for language generation, and "worlds" for video modeling. These abstractions serve as information bottlenecks that capture essential structure while enabling more efficient learning and better generalization. The research program spans four aims that develop this idea across different domains, from molecular design to video generation, creating a comprehensive approach to abstraction-enhanced artificial intelligence. The key innovation lies in treating these abstractions as instance-specific parameters that can be rapidly optimized during inference, similar to how biological systems combine fast episodic learning with slower semantic learning. This approach bridges modern deep learning with classical ideas about symbolic reasoning while maintaining the powerful function approximation capabilities of neural networks.

Intellectual Merit:

The proposed research advances artificial intelligence through several key contributions: (1) Development of a novel framework that unifies different forms of latent abstraction across multiple domains, from molecular design to video generation; (2) Development of algorithms and code for learning and inferring abstract representations that guide generation and optimization; (3) Providing insights into the relationship between memorization-based learning and abstraction-based learning and planning; (4) Demonstration of improved data efficiency and generalization through the use of explicit abstractions; (5) Creation of bridges between modern deep learning methods and classical ideas about symbolic reasoning and planning. The project introduces mathematically rigorous approaches to learning and using abstractions while maintaining the powerful function approximation capabilities of current methods.

Broader Impacts:

This project has significant potential impacts across multiple domains. The research will advance scientific discovery through better molecular and material design methods, improve robotics and automation through enhanced planning and control capabilities, enable more structured and logical natural language processing, and advance computer vision through physically-consistent video generation. The project will contribute substantially to workforce development through the training of graduate students and creation of new course materials on abstraction-based AI. A particular focus will be placed on providing research opportunities for underrepresented groups in STEM, including women and minorities, through targeted mentorship and inclusion in research projects. The educational impact extends beyond graduate education to include undergraduate research opportunities and the development of new curriculum materials that make advanced AI concepts accessible to a broader audience. The research will be disseminated through open-source software, publications, and educational materials, ensuring broad access to the developed methods and insights. Through collaborations with experimental laboratories and industry partners, the project ensures that theoretical advances translate into practical applications that benefit society. The combination of methodological innovation, educational development, and practical application creates a comprehensive program that advances both the field of artificial intelligence and its broader societal impact.

Latent Abstraction for Optimization and Generation

PI: Ying Nian Wu, UCLA Department of Statistics and Data Science

1 Introduction

The past decade has witnessed unprecedented success in artificial intelligence, driven by the remarkable capabilities of deep learning systems. At the heart of this revolution lies a powerful paradigm: over-parametrized networks that excel at memorization and interpolation. Large language models like GPT [6] have demonstrated extraordinary abilities in text generation through pure autoregressive prediction. Similarly, diffusion models [87, 33, 12] have revolutionized image and video generation, producing increasingly realistic outputs. However, these systems often require massive datasets and computation, struggle with systematic generalization, and lack explicit mechanisms for abstraction and reasoning.

This technological parallel recalls a profound transition in scientific history. In ancient astronomy, Ptolemy’s system of epicycles achieved remarkable predictive accuracy through increasingly complex combinations of circular motions. While effective for prediction, it lacked the profound conceptual understanding that Newton later provided through abstract concepts like force (F) and mass (m), crystallized in equations like $F = ma$. Newton’s theory, built on latent abstractions, offered not just prediction but deeper understanding and greater generalization. We propose that modern AI systems can benefit from a similar transformation through explicit latent abstractions.

Our framework introduces four manifestations of latent abstraction, each supported by strong preliminary results: (1) Latent “designs” for optimization problems like molecular design, where abstract representations capture chemical and functional properties while improving optimization efficiency; (2) Latent “plans” for sequential decision making, enabling more coherent long-term behavior in robotics through explicit planning; (3) Latent “thoughts” for language generation, providing mechanisms for more structured and logical text generation with enhanced consistency; (4) Latent “worlds” for video modeling, capturing physical dynamics and scene structure for better prediction and generation. These abstractions serve as information bottlenecks that capture essential structure while enabling more efficient learning and better generalization.

Our research program spans four aims that develop this unified framework across different domains, from molecular design to video generation. The key innovation lies in treating these abstractions as instance-specific parameters that can be rapidly optimized during inference, similar to how biological systems combine fast episodic learning with slower semantic learning [61]. This approach bridges modern deep learning with classical ideas about symbolic reasoning while maintaining the powerful function approximation capabilities of neural networks. Through comprehensive methodological development and empirical validation, we aim to establish a new paradigm for AI systems that combine the flexibility of deep learning with the power of explicit abstraction.

Computational Resources and Technical Expertise. The PI maintains a well-equipped research laboratory with significant computational resources, including **25 NVIDIA A6000 GPUs** for large-scale machine learning experiments. Through strategic collaboration with Dr. Jianwen Xie at Lambda, the team has access to **20 NVIDIA H100 GPUs**, with potential for additional resources as needed. The lab’s technical capabilities are further strengthened by Ph.D. students with expertise in GPU optimization and CUDA programming, including Deqian Kong and Andrew Lizarraga.

1.1 Intellectual Merit

The proposed research advances the methodological foundations and practical capabilities of artificial intelligence in several key ways: (1) Introduces a novel framework unifying different forms of latent abstractions across multiple domains. (2) Develops algorithms and code for learning and inference with abstract representations. (3) Provides insights into the relationship between memorization-based and abstraction-based learning. (4) Demonstrates practical benefits in terms of data efficiency, generalization, and interpretability. (5) Creates bridges between modern deep learning and classical ideas about symbolic reasoning and planning.

This work has the potential to significantly impact how we approach artificial intelligence, moving beyond pure prediction toward systems that combine the power of deep learning with explicit abstraction.

2 Latent Abstraction Framework

We propose a unified framework for learning and inference with latent abstractions that captures fundamental processes in both biological and artificial intelligence.

2.1 Notation and Problem Setup

Let $\mathbf{x} = (x^{(1)}, \dots, x^{(t)}, \dots, x^{(T)})$ denote an observed sequence, where $x^{(t)}$ represents an element at time step t , and T is the sequence length which may vary across different sequences. Let $\mathbf{z} \in \mathbb{R}^d$ denote the latent abstraction, which can take the form of a continuous vector or multiple vectors (sometimes referred to as “tokens” even when they are not quantized). When relevant, let $y \in \mathbb{R}$ denote a scalar value associated with the sequence \mathbf{x} , such as objective function value or cumulative reward, or $y \in \{0, 1\}$ denote binary value for task completion, answer correctness, constraint satisfaction, or human preference.

2.2 Forms of Latent Abstraction

The latent abstraction \mathbf{z} takes on different interpretations depending on the domain, revealing fundamental commonalities across various forms of intelligence:

Abstract “Design”. In molecular and biological design, \mathbf{x} naturally takes a sequential form. A molecule, though structurally a graph, can be encoded as a sequence through representations like SMILES [96] or SELFIES [51] notation. A protein is inherently sequential, represented as a chain of amino acids. Here, \mathbf{z} serves as an abstract “design” that captures essential structural and functional properties. The value y represents chemical or biological properties of interest, such as binding affinity, drug-likeness, protein stability, or enzyme activity. y can also be binary for constraint satisfaction. In multi-objective scenarios, y can be extended to a weighted combination of multiple property values, allowing for the optimization of complex design criteria while maintaining chemical and biological validity.

Abstract “Plan”. In reinforcement learning, where \mathbf{x} is a sequence of actions, \mathbf{z} functions as an abstract “plan” that guides behavior. This plan encodes high-level strategies and intentions, similar to how biological organisms formulate and execute complex action sequences. The value y represents the cumulative reward obtained from executing the action sequence, providing a measure of the plan’s effectiveness in achieving desired goals. This reward signal shapes the learning of effective planning strategies through experience. y can also be binary for task completion or game winning or losing.

Abstract “World”. For video sequences where \mathbf{x} consists of consecutive frames, possibly accompanied by self-motion actions in navigation or manipulation tasks, \mathbf{z} represents an abstract “world” or “map”. This internal model captures the environment’s structure and dynamics, enabling prediction and planning. The value y in this context can represent task completion (e.g., reaching a target location in navigation, successfully manipulating an object) or other measures of task success. Like cognitive maps in biological systems [93], this abstract representation helps in understanding spatial relationships and predicting future states.

Abstract “Thought”. When \mathbf{x} is a sequence of words or tokens, \mathbf{z} serves as an abstract “thought” that captures the underlying semantic content and structure. This parallels how biological intelligence might maintain abstract mental representations that guide language production and comprehension. In this setting, y can represent various measures of success: human preference scores for generated text, accuracy in reaching correct answers for question-answering tasks, or task completion metrics for instruction following. These values guide the model toward generating coherent, meaningful, and task-appropriate language.

2.3 Nature of Latent Abstraction

Latent Variables and Random Effect Parameters. The concepts of latent variables and random effect parameters, while arising from different modeling traditions [24], are fundamentally unified in their role

as instance-specific abstractions. Both serve the same essential purpose: they provide instance-specific parameters that enable adaptation to individual examples while sharing statistical strength across the dataset. This unity is particularly evident in our framework, where the latent abstraction z can be viewed either as latent variables encoding abstract properties or as random effect parameters capturing instance-specific structure.

Distinction from State Vector. The latent abstraction z in our framework warrants careful distinction from other forms of latent representations in sequential modeling. Unlike hidden state vectors s_t in state space models or recurrent neural networks, z represents a global abstraction of the entire sequence x . Our framework can be naturally extended to incorporate both levels of abstraction, where z first generates a sequence of hidden states ($s_t, t = 1, \dots, T$), which then generate the observations $x^{(t)}$ conditioned on both s_t and z .

Non-Sequential Data. While our initial formulation focuses on sequential data due to its widespread applicability and inherent temporal structure, the framework extends naturally to non-sequential data such as images and graphs.

2.4 Generative Model Formulation

We formulate our framework as a structured probabilistic model that captures the relationship between latent abstractions, observed sequences, and objective values.

Prior Model. Let $p_\alpha(z)$ denote the prior distribution of the latent abstraction z with parameter α . This prior can be realized in several ways. A particularly convenient formulation is through a transport model:

$$z = U_\alpha(z_0), \quad z_0 \sim \mathcal{N}(0, I), \quad (1)$$

where $U_\alpha(\cdot)$ is a learnable transformation. This transformation can be parametrized by a U-Net architecture [82] or, notably, by a transformer encoder [94]. In the latter case, our model becomes a repurposing of the original transformer architecture for translation [94], where the encoder maps Gaussian noise tokens z_0 to abstract tokens z that guide sequence generation. Alternative formulations include energy-based models [54] or diffusion models [33] for the prior.

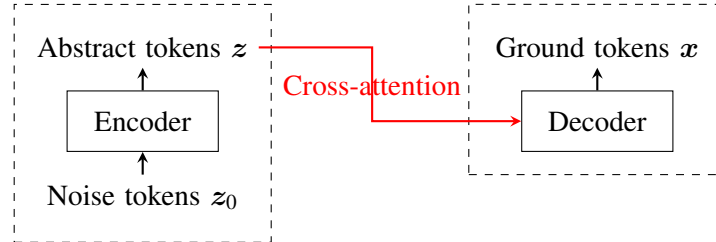


Figure 1: The architecture follows the original transformer encoder-decoder for translation, where the encoder serves as the prior for latent abstract token, and the decoder serves as the generator. Inference of latent tokens can be accomplished by inferring noise tokens.

Abstraction-Controlled Generation. The key component of our model is a z -controlled autoregressive generator $p_\beta(x|z)$ with parameter β . Unlike standard autoregressive models that only condition on previous elements [78], our model incorporates the global abstraction z at each generation step:

$$p_\beta(x|z) = \prod_{t=1}^T p_\beta(x^{(t)}|z, x^{(<t)}), \quad (2)$$

where $x^{(<t)}$ denotes previous tokens before $x^{(t)}$. We are particularly interested in models with a **finite context window** of size k : $p_\beta(x|z) = \prod_{t=1}^T p_\beta(x^{(t)}|z, x^{(t-k:t-1)})$, where $x^{(t-k:t-1)}$ denotes the k previous elements. This finite context forces z to serve as a global information carrier, integrating information across temporal segments that would otherwise be disconnected due to the limited context window.

For discrete elements $x^{(t)}$, we implement $p_\beta(x|z)$ using a transformer decoder where z cross-attends to the decoder layers. For image frames $x^{(t)}$, we can either employ diffusion models [12] or first quantize image patches using a VQ-GAN tokenizer [15] to maintain the discrete sequence modeling framework.

Value Prediction. When modeling sequences with associated value y , we introduce a prediction model $p_\gamma(y|z)$ with parameter γ , typically parametrized as a non-linear regression using a multi-layer perceptron:

$$p_\gamma(y|z) = \mathcal{N}(f_\gamma(z), \sigma^2), \quad (3)$$

where $f_\gamma(\cdot)$ is the prediction network and σ^2 is either fixed or learned. If $y \in \{0, 1\}$, then $p_\gamma(y|z)$ can be modeled by logistic regression.

Information Bottleneck. A crucial assumption in our model is that given z , the sequence x and value y are conditionally independent. This positions z as an information bottleneck [91], making it an objective-aware or value-aware abstraction. The joint distribution is thus: $p_\theta(x, y) = \int p_\beta(x|z)p_\gamma(y|z)p_\alpha(z)dz$, where $\theta = (\alpha, \beta, \gamma)$ denotes all model parameters.

2.5 Learning and Inference

We present three approaches for learning and inference in our framework, each offering different trade-offs between computational efficiency and modeling flexibility.

Maximum Likelihood Learning with Langevin Sampling. In this approach, we directly maximize the log-likelihood $L(\theta) = \frac{1}{n} \sum_{i=1}^n \log p_\theta(x_i, y_i)$. With the transport model prior $z = U_\alpha(z_0)$, we can write the joint distribution as:

$$p_\theta(x, y) = \int p_\beta(x|z = U_\alpha(z_0))p_\gamma(y|z = U_\alpha(z_0))p_0(z_0)dz_0, \quad (4)$$

where $p_0(z_0) = \mathcal{N}(0, \mathbf{I})$. The learning gradient is:

$$\nabla_\theta \log p_\theta(x, y) = \mathbb{E}_{p_\theta(z_0|x, y)}[\nabla_\theta \log p_\beta(x|U_\alpha(z_0)) + \nabla_\theta \log p_\gamma(y|U_\alpha(z_0))]. \quad (5)$$

The posterior distribution $p_\theta(z_0|x, y)$ is sampled using Langevin dynamics:

$$z_0^{\tau+1} = z_0^\tau + s \nabla_{z_0} \log p_\theta(z_0|x, y) + \sqrt{2s} \epsilon^\tau, \quad (6)$$

where τ indexes the time step, s is the step size, and $\epsilon^\tau \sim \mathcal{N}(0, \mathbf{I})$. The gradient term expands as:

$$\nabla_{z_0} \log p_\theta(z_0|x, y) = -z_0 + \nabla_{z_0} \sum_{t=1}^T \log p_\beta(x^{(t)}|x^{(<t)}, U_\alpha(z_0)) + \nabla_{z_0} \log p_\gamma(y|U_\alpha(z_0)). \quad (7)$$

Classical Variational Learning. In this approach, we introduce a variational posterior $q(z_0|x, y) = \mathcal{N}(\mu, \text{diag}(\sigma^2))$ with local parameters (μ, σ^2) specific to each (x, y) pair [43]. We maximize the evidence lower bound (ELBO) [35]:

$$\mathcal{L}(\theta, \mu, \sigma^2) = \mathbb{E}_{q(z_0|x, y)}[\log p_\beta(x|U_\alpha(z_0)) + \log p_\gamma(y|U_\alpha(z_0))] - \text{KL}(q(z_0|x, y) || p_0(z_0)). \quad (8)$$

Variational Auto-encoder with Amortized Inference. As a baseline, we consider the VAE approach [46] that introduces an inference network $q_\phi(z_0|x, y)$ with global parameters ϕ to amortize the iterative inference computation in classical variational learning.

Comparison. Our empirical studies suggest that both maximum likelihood learning with Langevin dynamics and classical variational learning can be more effective than the VAE approach, which requires learning a potentially very big network for the inference model. Inference in both MCMC-based and classical variational approaches can be made efficient with finite-step algorithms with warm start from previous learning iteration.

3 Aim 1: Latent “Design” for Optimization

3.1 Problem Formulation

Let \mathcal{X} denote a large structured input space. In combinatorial optimization problems, \mathcal{X} is typically discrete with cardinality that grows exponentially with problem size. For instance, in molecular design, \mathcal{X} represents the space of valid molecules, where even modest-sized molecules with up to 40 atoms can yield a space of over 10^{60} possible combinations. Each $\mathbf{x} \in \mathcal{X}$ can be represented as a sequence $\mathbf{x} = (x^{(1)}, \dots, x^{(T)})$, such as SMILES [96] or SELFIES [51] notation for molecules.

The goal is to find $\mathbf{x}^* \in \mathcal{X}$ that maximizes an objective function $F : \mathcal{X} \rightarrow \mathbb{R}$ [5]: $\mathbf{x}^* = \arg \max_{\mathbf{x} \in \mathcal{X}} F(\mathbf{x})$. In black-box optimization, $F(\mathbf{x})$ is unknown but can be evaluated through queries to an oracle function. The oracle returns a value $y = F(\mathbf{x})$ for any input \mathbf{x} . In molecular design, oracle functions include computational chemistry tools like RDKit [53] for calculating drug-likeness (QED) [3] and synthetic accessibility (SA) [14], AutoDock-GPU [83] for protein binding affinity, wet-lab measurements through surface plasmon resonance (SPR) [75], or trained surrogate models that predict properties [26].

Starting from an initial training dataset $\mathcal{D} = (\mathbf{x}_i, y_i)_{i=1}^n$ of input-output pairs, where $y_i = F(\mathbf{x}_i)$, we aim to find optimal \mathbf{x}^* . In **offline optimization**, we must rely solely on the initial dataset without additional oracle queries [16]. **Online optimization** allows continued interaction with the oracle to evaluate new candidates, enabling iterative improvement through active learning [84]. For online optimization, the initial dataset \mathcal{D} can use randomly generated (\mathbf{x}_i) . This proposal focuses on online optimization where we can continue to query the oracle function.

Traditional approaches to black-box optimization often rely on Bayesian optimization with Gaussian processes as surrogate models [79, 86]. However, Gaussian processes scale poorly to high-dimensional structured inputs like molecules. Recent work has shown that generative models can effectively learn the distribution of high-performing inputs $p(\mathbf{x}|\mathbf{y})$ [41, 104], enabling more efficient optimization in large combinatorial spaces [110].

3.2 Proposed Framework on Learning and Distribution Shifting

We can pretrain the model $p_\theta(\mathbf{x}, \mathbf{y}, \mathbf{z})$ with transport prior $\mathbf{z} = U_\alpha(\mathbf{z}_0)$, $\mathbf{z}_0 \sim \mathcal{N}(0, \mathbf{I})$ using the initial dataset \mathcal{D} . After training, optimization can be accomplished through conditional generation: sampling $\mathbf{z}_0 \sim p_\theta(\mathbf{z}_0|\mathbf{y}) \propto p_0(\mathbf{z}_0)p_\gamma(\mathbf{y}|\mathbf{z} = U_\alpha(\mathbf{z}_0))$ using Langevin dynamics, followed by generating $\mathbf{x} \sim p_\beta(\mathbf{x}|\mathbf{z} = U_\alpha(\mathbf{z}_0))$. The σ^2 parameter in $p_\gamma(\mathbf{y}|\mathbf{z}) = \mathcal{N}(f_\gamma(\mathbf{z}), \sigma^2)$ controls exploration (for big σ^2) and exploitation (for small σ^2).

Naively setting \mathbf{y} to a high target value leads to unreliable extrapolation beyond the training distribution. We propose a gradual shifting scheme that iteratively moves the learned distribution toward regions of higher objective values. The algorithm maintains a replay buffer \mathbf{B}^t of size K , initialized by selecting the top- K molecules from the training dataset \mathcal{D} based on their objective values. At iteration t , for each example (\mathbf{x}, \mathbf{y}) in \mathbf{B}^t , we shift its objective value by $\hat{\mathbf{y}} = \mathbf{y} + \Delta$, where Δ is a small step size. For each shifted target $\hat{\mathbf{y}}$, we generate M new molecules by sampling latent vectors using the posterior $p_\theta(\mathbf{z}_0|\hat{\mathbf{y}})$, then generating molecules from $p_\beta(\mathbf{x}|\mathbf{z} = U_\alpha(\mathbf{z}_0))$. Each generated molecule is evaluated by the oracle to obtain its true value $\mathbf{y} = F(\mathbf{x})$.

Let \mathbf{G}^t denote the set of generated molecules and their oracle values. The new buffer is formed by selecting the top- K molecules based on objective values: $\mathbf{B}^{t+1} = \text{TopK}(\mathbf{B}^t \cup \mathbf{G}^t)$. The model parameters are then updated through maximum likelihood training on \mathbf{B}^{t+1} . This process continues until convergence or until the oracle query budget is exhausted.

The gradual shifting ensures reliable extrapolation by maintaining the model’s support near observed data while progressively moving toward regions of higher objective values. The replay buffer serves both as a mechanism for sample-efficient learning and as a way to prevent catastrophic forgetting of previously discovered high-performing molecules.

Computational Efficiency. For both posterior sampling and model learning, we adopt finite-step Langevin dynamics with warm start from previous learning or shifting iteration. Our empirical studies

show that as few as 2-3 Langevin steps per iteration can be sufficient when combined with warm start.

3.3 Preliminary Results on Molecule Design

Our preliminary work, reported in our NeurIPS 2024 paper [47], demonstrates the effectiveness of latent abstraction for molecule design through a comprehensive evaluation framework encompassing both multi-objective optimization and targeted drug design. In multi-objective optimization, we simultaneously optimize binding affinity (K_D), drug-likeness (QED), and synthetic accessibility (SA) across three protein targets: ESR1, ACAA1, and PHGDH [83]. The binding affinity is expressed as the dissociation constant K_D (nM), where lower values indicate stronger binding, computed using AutoDock-GPU as a proxy for experimental measurements. As shown in Table 1, our method achieves superior binding affinity while maintaining favorable drug-like properties across all targets.

Table 1: Multi-objective optimization results. Top 2 performance, measured by K_D (nM), QED and SA, are reported for each method. Baseline methods include LIMO [13] and SGDS [48]. Best results are marked in bold, and the second best results are underlined.

| LIGAND | ESR1 | | | ACAA1 | | | PHGDH | | |
|---------------------------|------------------|-------------|-------------|------------------|-------------|-------------|------------------|-------------|-------------|
| | K _D ↓ | QED ↑ | SA ↓ | K _D ↓ | QED ↑ | SA ↓ | K _D ↓ | QED ↑ | SA ↓ |
| LIMO 1 ST | 4.6 | 0.43 | 4.8 | 28 | 0.57 | 5.5 | 29.15 | 0.33 | 4.73 |
| LIMO 2 ND | 2.8 | 0.64 | 4.9 | 31 | 0.44 | 4.9 | 42.98 | 0.20 | 5.32 |
| SGDS 1 ST | 0.36 | 0.44 | 3.99 | 4.55 | 0.56 | 4.07 | 4.47 | 0.54 | 3.37 |
| SGDS 2 ND | 1.28 | 0.44 | 3.86 | 5.67 | <u>0.60</u> | 4.58 | 5.39 | 0.42 | 4.02 |
| LPT 1ST | 0.04 | <u>0.58</u> | <u>3.46</u> | 0.18 | 0.50 | 4.85 | 0.02 | <u>0.50</u> | 3.11 |
| LPT 2ND | <u>0.05</u> | 0.46 | 3.24 | <u>0.21</u> | 0.61 | <u>4.18</u> | <u>0.03</u> | 0.43 | <u>3.22</u> |

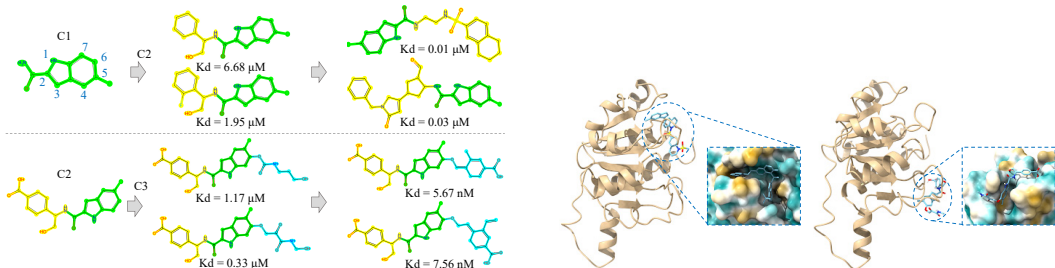


Figure 2: Left: Structure-constrained optimization starting from an indole backbone (C1). Left column shows our model’s ability to reproduce expert-designed compounds C2 and C3, while right column shows novel compounds with improved binding affinity. Right: Binding poses visualized in PHGDH’s NAD binding site, showing how our generated molecules (left: multi-objective optimization, right: structure-constrained optimization) maintain key protein-ligand interactions.

We further demonstrate our method’s capabilities through a detailed case study on Phosphoglycerate dehydrogenase (PHGDH), an enzyme pivotal in L-serine synthesis that has emerged as a promising therapeutic target in cancer treatment [64]. PHGDH is particularly relevant as its overexpression has been linked to breast cancer progression and poor patient outcomes [77]. For PHGDH inhibitor design, we employ structure-constrained optimization starting from an indole backbone scaffold, which has been previously validated through experimental studies [95].

As shown in Figure 1(left), our method successfully reproduces the progression of human expert designs (compounds $C1 \rightarrow C2 \rightarrow C3$) while also discovering novel variants with improved predicted binding affinity. Figure 1(right) visualizes how our generated molecules maintain key interactions with the NAD binding site while achieving stronger predicted binding. These results demonstrate our approach’s ability to effectively navigate the chemical space around a given scaffold while maintaining chemical and structural validity.

3.4 Proposed Project 1.1: Optimization of PHGDH Inhibitors

We propose to continue our work on designing PHGDH inhibitors through an established collaboration with Prof. Sheng Zhong’s laboratory at UCSD (co-authors in our NeurIPS 2024 paper on molecule design [47]). While our preliminary work demonstrates strong computational results, the ultimate validation of our approach requires experimental verification of the generated molecules. The Zhong lab has extensive experience in PHGDH biochemistry and cancer metabolism [80], making them ideal collaborators for this project.

The project has three main components. First, we will expand our structure-constrained optimization to incorporate more sophisticated chemical constraints derived from known PHGDH inhibitors [63] and medicinal chemistry principles [40]. Specifically, we will develop a pharmacophore-guided generation approach to ensure that our generated molecules maintain key binding features while exploring novel chemical space.

Second, we will validate our approach through experimental testing in the Zhong lab. Selected molecules will be synthesized and tested for: (1) binding affinity through surface plasmon resonance (SPR) measurements [75], (2) inhibition of PHGDH enzymatic activity through biochemical assays [68], and (3) efficacy in cancer cell lines known to be dependent on PHGDH activity [77]. The experimental results will be used to refine our computational model in an iterative manner, creating a tight feedback loop between computational design and experimental validation.

3.5 Proposed Project 1.2: Dual Space Optimization and Search

For the objective function $F(\mathbf{x})$, either given or learned from training data, our latent abstraction framework offers a unique advantage by enabling gradient-based search in continuous latent space \mathbf{z} . Given a target value y , we can efficiently sample $\mathbf{z}_0 \sim p_\theta(\mathbf{z}_0|y)$ using Langevin dynamics guided by $\nabla_{\mathbf{z}_0} \log p_\gamma(y|\mathbf{z} = U_\alpha(\mathbf{z}_0))$. This avoids the need for ad hoc discrete search heuristics in the original space \mathcal{X} . The learned generative model $p_\beta(\mathbf{x}|\mathbf{z})$ can then produce novel solutions that interpolate and extrapolate from training examples.

We propose to combine this latent space approach with direct search in data space \mathcal{X} . One example of data space search is through MCMC sampling of the Gibbs/Boltzmann distribution: $p(\mathbf{x}) = \frac{1}{Z_T} \exp(F(\mathbf{x})/T)$, where T is a temperature parameter, controlling exploration and Z_T is the normalizing constant. We shall also explore other search heuristics such as A* [32] and its variants [36].

We plan to evaluate this dual space approach on two optimization problems that are very appealing to the PI: (1) Protein design, where \mathbf{x} represents amino acid sequences optimized for specific functions such as enzyme catalysis [39] and antibody-antigen binding [60]. In fact, finding the latent abstraction \mathbf{z} for protein sequence is itself can be useful for understanding the protein and for predicting its folding. (2) Search problems in automated theorem proving, where \mathbf{x} represents proof steps in formal mathematics [76, 29]. These domains feature exponentially large discrete search spaces with complex structure and constraints, making them ideal testbeds for our approach.

The key insight is that latent space sampling provides global search capabilities by operating in a continuous, learned abstraction space that captures essential problem structure, while data space search enables precise local refinement by directly optimizing the objective function. This complementarity suggests that combining both strategies may lead to more effective optimization algorithms that balance exploration and exploitation in complex combinatorial spaces.

4 Aim 2: Latent “Plan” for Reinforcement Learning and Robotics

4.1 Problem Formulation

Modern reinforcement learning has achieved remarkable success through the use of step-wise rewards and value function estimation [89]. However, designing effective reward functions requires significant domain expertise and often fails to capture true task objectives. This becomes particularly challenging in real-world robotics applications where natural tasks rarely provide immediate feedback.

Table 2: Evaluation results of offline OpenAI Gym MuJoCo tasks. We provide results for data specification with step-wise reward (left) and final return (right). **Bold** highlighting indicates top scores. LPT outperforms all final-return baselines and most step-wise-reward baselines.

| Dataset | Step-wise Reward | | | Final Return | | | | |
|---------------------------|------------------|------|------|--------------|------|------|------------------|-------------------------|
| | CQL | DT | QDT | CQL | DT | QDT | LPT (Ours) | LPT-EI (Ours) |
| halfcheetah-medium | 44.4 | 42.1 | 42.3 | 1.0 | 42.4 | 42.4 | 43.13 \pm 0.38 | 43.53 \pm 0.08 |
| halfcheetah-medium-replay | 46.2 | 34.1 | 35.6 | 7.8 | 33.0 | 32.8 | 39.64 \pm 0.83 | 40.66 \pm 0.12 |
| hopper-medium | 58.0 | 60.3 | 66.5 | 23.3 | 57.3 | 50.7 | 58.52 \pm 1.92 | 63.83 \pm 1.47 |
| hopper-medium-replay | 48.6 | 63.7 | 52.1 | 7.7 | 50.8 | 38.7 | 82.29 \pm 1.26 | 89.93 \pm 0.61 |
| walker2d-medium | 79.2 | 73.3 | 67.1 | 0.0 | 69.9 | 63.7 | 77.85 \pm 3.18 | 81.15 \pm 0.33 |
| walker2d-medium-replay | 26.7 | 60.2 | 58.2 | 3.2 | 51.6 | 29.6 | 72.31 \pm 1.92 | 75.68 \pm 0.34 |
| kitchen-mixed | 51.0 | 22.3 | - | - | 17.2 | - | 61.9 \pm 1.22 | 64.7 \pm 0.51 |
| kitchen-partial | 49.8 | 20.4 | - | - | 10.5 | - | 61.2 \pm 1.75 | 65.3 \pm 0.62 |

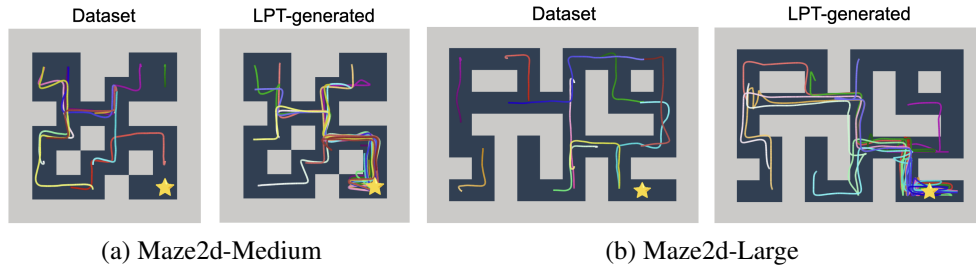


Figure 3: (a) Maze2D-medium environment (b) Maze2D-large environment. Left panels show example trajectories from the training set and right panels show LPT generations. Yellow stars represent the goal states.

We propose to fundamentally rethink reinforcement learning through the lens of latent abstractions. Instead of relying on hand-crafted step-wise rewards, we focus on learning from trajectory-return pairs (τ, y) , where $\tau = (x^{(1)}, \dots, x^{(T)})$ represents a trajectory of states and actions, and y represents the total return or task success metric. This formulation naturally aligns with how humans specify goals and evaluate success in real-world tasks.

The key innovation is introducing a latent plan z that serves as an abstract representation connecting trajectory generation to final outcomes. This plan should exhibit three essential properties: (1) significance - prioritizing more important returns, (2) persistence - maintaining consistency even when outcomes are determined in hindsight, and (3) contingency - adapting to environmental changes during execution. We formulate this mathematically as a structured probabilistic model: $p_{\theta}(\tau, y, z) = p_{\alpha}(z)p_{\beta}(\tau|z)p_{\gamma}(y|z)$, where $p_{\alpha}(z)$ is a learnable prior over latent plans, $p_{\beta}(\tau|z)$ generates trajectories conditioned on the plan, and $p_{\gamma}(y|z)$ predicts expected returns. This formulation positions planning as inference in latent space: given a desired return, the agent infers a latent plan through posterior sampling, then uses this plan to guide trajectory generation.

4.2 Preliminary Results

In our NeurIPS 2024 paper on latent plan transformer [50], we have evaluated our latent plan approach on offline reinforcement learning tasks [55] (i.e., learning from a fixed training dataset without additional environment interaction) across a comprehensive suite of environments: OpenAI Gym-Mujoco [92] for continuous control, Franka Kitchen [27] for robotic manipulation, Maze2D [17] for navigation, and Connect Four [85] for contingent planning. The data specification of trajectory-return pairs, without step-wise rewards, distinguishes our study from most existing work in reinforcement learning.

On the Gym-Mujoco continuous control benchmarks (Table 2), our method matches or exceeds the performance of methods that rely on dense reward signals [18], even without access to step-wise rewards. This demonstrates effective credit assignment over long action sequences.

Table 3: Evaluation results on Connect Four. **Bold** highlighting indicates top scores.

| Dataset | CQL | DT | ESPER | LPT |
|--------------|-----------------|----------------|-----------------------------------|-----------------------------------|
| Connect Four | 0.61 ± 0.05 | 0.8 ± 0.07 | 0.99 ± 0.03 | 0.99 ± 0.01 |

The ability to compose partial solutions, termed trajectory “stitching”, is demonstrated in the Maze2D experiments (Figure 3). Despite training data containing mainly suboptimal trajectories, our method can generate successful goal-reaching behaviors by effectively combining trajectory segments.

Finally, to evaluate robustness to environmental stochasticity, we tested our approach on Connect Four where an agent must adapt to adversarial opponent moves. As shown in Table 3, our method achieves near-perfect performance (0.99 ± 0.01), significantly outperforming traditional approaches and matching the state-of-the-art ESPER method [2]. It demonstrates that the latent plan representation effectively captures strategic understanding while maintaining adaptability to opponent actions.

4.3 Proposed Project 2.1: Online Learning through Gradual Distribution Shifting

The first project explores how latent plan inference can guide online exploration and adaptation. While our preliminary work demonstrates strong performance in offline settings, the ability to interact with the environment opens new possibilities for continual improvement. We propose to adapt the gradual distribution shifting strategy from molecular optimization to the online reinforcement learning setting.

The key idea is to maintain a replay buffer \mathbf{B}^t of size K , initialized with the top-performing trajectories from offline training data \mathcal{D} . At each iteration t , for each trajectory-return pair (τ, y) in \mathbf{B}^t , we shift the target return by $\hat{y} = y + \Delta$, where Δ is a small step size. For each shifted target, we generate new trajectories by sampling latent plans $z_0 \sim p_\theta(z_0|\hat{y})$ using Langevin dynamics, executing these plans in the environment to collect actual trajectories and returns, and adding successful trajectories to a generated set \mathbf{G}^t . The buffer is updated by selecting the top- K trajectories from the union: $\mathbf{B}^{t+1} = \text{TopK}(\mathbf{B}^t \cup \mathbf{G}^t)$. The model parameters are then updated through maximum likelihood training on \mathbf{B}^{t+1} .

This process iteratively shifts the distribution toward higher-return regions while maintaining reliable generation through the replay buffer mechanism. Our preliminary online learning experiments show promise on the D4RL benchmarks [17]. On the Walker2d environment, this approach improves performance from 72.31 to 78.99, and on Hopper from 82.29 to 89.93, demonstrating effective distribution shift.

4.4 Proposed Project 2.2: Adaptive Re-planning through Sequential Latent Updates

The second project explores how latent plans can be continuously refined during execution, inspired by both model predictive control in engineering and adaptive planning in biological systems. Rather than inferring a complete, precise plan before execution, we propose to start with a rough plan that is iteratively refined as the agent interacts with the environment.

At each time step t , given the current state s_t and executed trajectory prefix $\tau_{1:t} = (s_1, a_1, \dots, s_t)$, we update the latent plan z using a small number of Langevin steps. Crucially, we only need a few steps ($k = 2$ or 3) at each time step, as the plan only needs incremental refinement based on new information. This approach offers several advantages: (1) computational efficiency through incremental updates rather than complete re-planning, (2) robustness to modeling errors and environmental uncertainty through continuous adaptation, and (3) biological plausibility as it mirrors how animals refine their plans during execution.

Current Progress: Through an ongoing collaboration with Prof. Denis Hong’s Robotics and Mechanisms Laboratory (RoMeLa) at UCLA Electrical Engineering, we have obtained promising preliminary results for box-catching tasks using our latent plan framework in their custom-designed robotic simulation environment, which minimizes the sim-to-real gap. Our method achieves real-time replanning and outperforms diffusion model baselines.

5 Aim 3: Latent “Thought” for Language Generation

5.1 Problem Formulation

Let $\mathbf{x} = (x^{(1)}, \dots, x^{(T)})$ be a sequence of ground tokens and \mathbf{z} be the latent abstract tokens that guide generation. Modern language models have achieved remarkable success through pure autoregressive prediction $p(x^{(t)} | x^{(<t)})$ [6], but this approach has three key limitations: (1) lack of explicit abstraction mechanisms analogous to human thought, making models less data efficient than human learners, (2) inability to perform significant computation or inference at test time, instead relying solely on knowledge baked into parameters during training, and (3) difficulty in maintaining long-term coherence and logical consistency across long sequences.

We propose to address these limitations through a structured probabilistic model $p_{\theta}(\mathbf{x}, \mathbf{z}) = p_{\alpha}(\mathbf{z})p_{\beta}(\mathbf{x}|\mathbf{z})$, where $p_{\alpha}(\mathbf{z})$ is implemented by mapping Gaussian noise $\mathbf{z}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ through a transformer encoder [94], and $p_{\beta}(\mathbf{x}|\mathbf{z})$ is a transformer decoder where \mathbf{z} cross-attends to guide generation.

The learning process employs classical variational Bayes [4] with dual learning rates: fast optimization of local variational parameters (μ, σ^2) specific to each sequence using approximately 20 steps of Adam [45], and slow optimization of global parameters $\theta = (\alpha, \beta)$ shared across all sequences. This mirrors the distinction between rapid episodic learning and gradual semantic learning in human cognition [52], while avoiding the need for a separate inference network as in VAEs [46].

5.2 Preliminary Results

We evaluated our approach on OpenWebText [25], a large-scale text corpus created by scraping and filtering Reddit submissions. OpenWebText contains approximately 8 million documents spanning diverse topics and writing styles, making it an excellent testbed for general language modeling capabilities [78]. Our experiments used a 500M parameter model to analyze both training efficiency and zero-shot generalization [6].

Table 4: Evaluation results of Gen PPL.

| Model | 13B | 22B | 28B | 100B |
|-------|--------|-------|--------------|--------|
| Ours | 145.83 | 98.07 | 87.17 | - |
| AR | 112.23 | 97.15 | 112.66 | 102.55 |

Table 5: Evaluation results of 0-shot PPL, compared to Diffusion and AR baselines.

| Task | Ours (28B) | SEDD | MDLM | AR |
|----------|--------------|--------|-------|-------|
| PTB | 37.70 | 100.09 | 95.26 | 82.05 |
| Wikitext | 14.65 | 34.28 | 32.83 | 25.75 |
| Lambada | 16.78 | 49.86 | 47.52 | 51.28 |
| Lm1b | 20.18 | 68.20 | 67.01 | 51.25 |

Our first set of experiments focused on analyzing the scaling behavior of Generative Perplexity (Gen PPL) [90] with training data size. The results, shown in Table 6, demonstrate the effectiveness of our latent abstraction approach compared to standard autoregressive (AR) models [94]. At 28B tokens of training data, our model achieves a Gen PPL of 87.17, significantly outperforming the AR baseline’s 112.66. While the AR baseline’s performance fluctuates (112.23 at 13B tokens, improving to 97.15 at 22B tokens, but degrading to 112.66 at 28B tokens) [44], our model shows steady improvement (145.83, 98.07, and 87.17 for 13B, 22B, and 28B tokens respectively). This consistent scaling behavior suggests that our explicit abstraction mechanism enables more stable learning from additional data.

To evaluate generalization capabilities, we conducted zero-shot evaluations [78] on four standard benchmarks, comparing against recent diffusion-based models (SEDD [56], MDLM [108]) and autoregressive baselines. As shown in Table 7, our model achieves superior performance across all tasks: (1) Penn Treebank (PTB) [59], a carefully annotated corpus of English text, (2) WikiText [62], derived from high-quality Wikipedia articles, (3) LAMBADA [73], testing long-range dependencies and coherence, and (4) Lm1b [9], based on news articles. Our model’s strong performance on LAMBADA (16.78 versus baselines > 47) is particularly noteworthy, suggesting that the latent abstractions effectively capture long-range dependencies and semantic relationships.

5.3 Proposed Project 3.1: Understanding Multi-layer Latent Abstractions

Our current work uses different latent tokens to attend to different layers of the transformer decoder, hypothesizing that this structure might naturally induce a hierarchy of abstractions. We propose to systematically investigate this hypothesis through rigorous analysis and experimental studies.

The research will proceed along three directions. First, we will analyze the attention patterns between latent tokens and decoder layers to understand the emergent division of labor. Preliminary observations suggest that lower-layer-attending tokens focus on local syntactic patterns, while higher-layer tokens capture broader semantic and structural relationships. However, a deeper understanding of this specialization is needed.

Second, we will develop methods to explicitly encourage hierarchical organization in the latent space. Building on cognitive theories of language processing, we propose that tokens attending to different decoder layers should capture distinct aspects: (1) low-level tokens encode lexical and syntactic information, (2) mid-level tokens represent semantic relationships and discourse structure, and (3) high-level tokens maintain global coherence and stylistic consistency. This can be achieved through carefully designed architectural constraints and training objectives.

Third, we will investigate the relationship between latent abstractions and human-interpretable linguistic concepts. By analyzing how different latent tokens influence generation across varying linguistic contexts, we can map the learned abstractions to established frameworks in linguistics and cognitive science. This includes studying how latent tokens guide phenomena like syntactic agreement, semantic roles, discourse coherence, and pragmatic inference.

5.4 Proposed Project 3.2: Dual Continuous-Symbolic Abstractions

We propose to extend our framework by augmenting the continuous latent tokens z with symbolic tokens $s = (s^{(1)}, \dots, s^{(m)})$ drawn from a small vocabulary. While z captures rich continuous representations, s enables explicit symbolic reasoning through a sequence of m discrete steps analogous to logical deduction.

The generative process follows a two-stage structure: $p(s|z^{(0)})p(x|z^{(1)}, s)$. In the first stage, $z^{(0)}$ generates a sequence of symbolic tokens s representing abstract logical steps. For example, s might capture reasoning patterns like “person 1 performs action A on object B, which causes person 2 to respond with action C.” The second stage uses $z^{(1)}$ to instantiate these abstract patterns with specific details, such as mapping “person 1” to “Tom Hanks” or “action A” to “offers assistance.” $z = (z^{(0)}, z^{(1)})$, where $z^{(0)}$ represents logical and reasoning patterns and $z^{(1)}$ represents concrete details for grounding.

For computational efficiency, we propose to first train a short-circuited model $p(x|z)$ using our established variational framework. This allows us to infer z efficiently through continuous optimization. With the initially inferred z , we can then update both s and z from the posterior $p(s, z|x)$ in the full model where s can be updated by discrete search. This approach combines efficient global exploration of continuous optimization with the interpretability and accurate local search of symbolic reasoning, similar to Proposed Project 1.2.

This hybrid architecture offers several advantages: (1) The symbolic sequence s provides an interpretable trace of the model’s reasoning process, (2) The vocabulary constraint on s encourages the learning of generalizable rules and patterns, (3) The continuous latent space z maintains the flexibility to handle nuanced variations and details not captured by the symbolic vocabulary. The research will focus on developing effective architectures for integrating continuous and symbolic representations, designing appropriate symbolic vocabularies, and evaluating the model’s ability to learn and apply logical rules in language generation.

Success in this project would bridge the gap between neural and symbolic approaches to language processing, potentially leading to models that combine the flexibility of deep learning with the interpretability and generalization capabilities of symbolic systems.

Finally we can incorporate y for correctness in question and answer or for human preference.

6 Aim 4: Latent “World” for Video Generation

6.1 Problem Formulation

The evolution of vision in biological systems marks one of the most significant developments in the history of life, contributing to the Cambrian explosion by enabling complex navigation and manipulation [74]. This suggests that understanding visual perception through the lens of navigation and action may be fundamental to advancing artificial intelligence.

Let $\mathbf{x} = (x^{(1)}, \dots, x^{(T)})$ denote a video sequence of frames, and $\boldsymbol{\tau} = (\tau^{(1)}, \dots, \tau^{(T)})$ represent the sequence of self-motion actions (e.g., camera movements or robotic controls). The key innovation of our approach is to introduce latent tokens \mathbf{z} that encode an abstract “world model” - capturing both the 3D structure of the environment and potential trajectories through it. When conditioning specifically on self-motion $\boldsymbol{\tau}$, the model $p_{\theta}(\mathbf{x}|\mathbf{z}, \boldsymbol{\tau})$ encourages \mathbf{z} to focus purely on encoding environmental structure, as motion information is provided explicitly.

This formulation fundamentally reframes video understanding as a navigation problem. Rather than treating each frame as an independent image to be processed, our model learns to extract abstract representations that support prediction of how the environment will appear from different viewpoints. This mirrors the primary evolutionary role of vision in biological systems - enabling organisms to navigate through and interact with their environment. The abstract tokens \mathbf{z} serve as a learned cognitive map [93], analogous to neural representations in the hippocampus that encode spatial structure and support planning [67].

We can further factorize the generative process as $p_{\theta}(\mathbf{x}, \boldsymbol{\tau}|\mathbf{z}) = p_{\alpha}(\boldsymbol{\tau}|\mathbf{z})p_{\beta}(\mathbf{x}|\mathbf{z}, \boldsymbol{\tau})$, where $p_{\alpha}(\boldsymbol{\tau}|\mathbf{z})$ generates plausible trajectories through the encoded environment [28] and $p_{\beta}(\mathbf{x}|\mathbf{z}, \boldsymbol{\tau})$ renders the corresponding visual observations. This decomposition enables both prediction of future frames given a planned trajectory and inference of environmental structure from observed sequences. The framework provides a unified approach to video modeling that emphasizes the fundamental relationship between vision, spatial understanding, and action.

This framework’s use of explicit latent abstractions is particularly natural for video modeling because visual sequences are inherently highly redundant and compressible.

6.2 Preliminary Results

We evaluated our approach on DMLab-40k [1], a dataset of procedurally generated 3D maze environments created using the DeepMind Lab simulator [42]. The dataset consists of 40,000 videos, each containing 300 frames at 64×64 resolution, showing an agent navigating through randomly generated 7×7 mazes with diverse floor and wall textures. The navigation paths are generated by selecting random target points in the maze and following the shortest route to reach them, providing a rich test bed for evaluating both environmental understanding and motion prediction.

Our model demonstrates strong capabilities in two key scenarios: action-conditioned prediction and unconditional generation. In action-conditioned prediction, where self-motion sequences $\boldsymbol{\tau}$ are provided, the latent tokens \mathbf{z} effectively capture the underlying maze structure, enabling accurate prediction of visual observations from novel viewpoints. The model maintains consistency in structural elements like walls and corridors while preserving textural details across long sequences of predicted frames.

For unconditional generation, where the model must generate both plausible trajectories and corresponding visual observations, we observe that the learned latent representations encode not only environmental structure but also typical patterns of agent movement through maze environments. The model generates coherent sequences that respect physical constraints of the environment, such as not passing through walls and maintaining consistent spatial relationships between different maze segments.

Qualitative analysis reveals that different components of the latent tokens \mathbf{z} specialize in representing distinct aspects of the environment: some encode persistent structural features like maze layout, while others capture variable elements like surface textures. This emergent specialization suggests that the model learns to decompose the scene into stable geometric structure and appearance variations, aligning with our goal of learning compressed abstract representations of the physical world.

6.3 Proposed Project 4.1: Scaling to Real-World Video Understanding

We propose to extend our latent world model framework from synthetic maze environments to real-world videos.

For videos with known camera motion τ (e.g., from robot platforms or stabilized handheld devices), we can directly apply our conditional model $p(x|z, \tau)$. The research will focus on developing architectures that allow z to capture hierarchical scene representations: from low-level geometry and appearance to high-level scene organization and object relationships. This structured representation should enable novel view synthesis and prediction of how scenes will appear from new viewpoints.

For videos without explicit motion information, we will explore two approaches. First, we can treat τ as a latent variable to be inferred alongside z , using our classical variational framework with fast inference of both scene structure and camera motion. Second, we can work directly with the unconditional model $p(x|z)$, where z must encode both environmental structure and typical patterns of motion through the scene. This latter approach may better capture how biological systems learn world models without access to explicit motion signals.

We will evaluate these approaches on several real-world video datasets: (1) indoor navigation sequences from the Gibson environment [97], which provides ground truth camera poses and photorealistic scans of over 1,400 real spaces for validation, (2) ego-centric videos from the EPIC-KITCHENS dataset [11], featuring complex manipulation activities and object interactions in dynamic kitchen environments, and (3) in-the-wild videos from the Kinetics-700 dataset [7], testing the model’s ability to handle unconstrained real-world scenes with diverse activities and environments. We can incorporate y for reaching the goal or completing the task.

The goal is to develop models that can form robust abstract representations of real environments, enabling both prediction and reasoning about spatial structure.

6.4 Proposed Project 4.2: Diffusion Models with Random Effect Parameters

We propose to explore an alternative formulation of latent abstractions using the parameters of Low-Rank Adaptation (LoRA) [38] as random effect parameters [23], which correspond to our z . While our primary framework treats latent variables z as explicit tokens, this project investigates how model parameters themselves can serve as instance-specific abstractions in diffusion models [87, 33, 88].

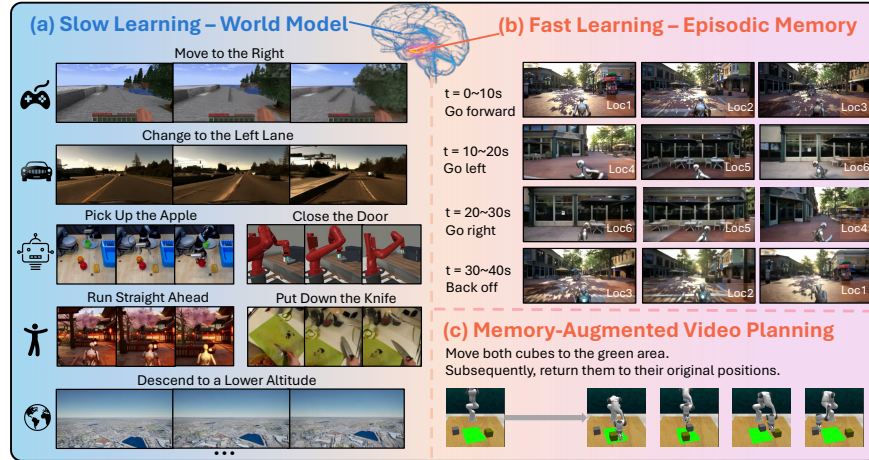


Figure 4: SLOWFAST-VGEN, a random effect text video generation system that mimics the complementary learning system in human brains. The slow learning phase (a) learns a model that simulates general dynamics across a diverse set of scenarios. The fast learning phase (b) stores episodic memory for consistent long video generation, e.g., generating the same scene for “Loc1” after traveling across different locations. Slow-fast learning also facilitates long-horizon planning tasks (c) that require the efficient storage of long-term episodic memories.

The key idea is to augment a diffusion model’s parameters θ with LoRA parameters in the form of low

rank matrices B and A that are inferred for each example or sequence. The diffusion process maintains its standard form, but the parameters $\theta + BA$ now include both global parameters θ shared across all examples and local parameters B and A specific to each instance. Inference of B and A follows our established dual-rate optimization: fast learning of LoRA parameters for each example using around 20 steps of Adam [45], and slow learning of global parameters across the dataset.

In our ICLR 2025 submission [37] (which has received high ratings from reviewers), we have obtained preliminary results for text to video generation, where the text input is in the form of description of actions. Fast learning of LoRA parameters enables generation of long videos. Figure 4 shows some generated video frames.

Success in this project would not only advance our understanding of parameter-based abstractions but also provide practical benefits in terms of more efficient and adaptable generative models. The research will focus on developing effective architectures for parameter inference [81], studying the relationship between LoRA rank and abstraction quality, and evaluating the approach on video generation tasks where temporal coherence is crucial [34].

7 Timeline and Milestones

Some of the preliminary results have been published in our recent NeurIPS 2024 papers [47, 50]. The proposed projects represent new directions beyond what we have published and submitted. The following table outlines timeline of our plan.

Table 6: Development stages of proposed projects.

| Projects | Year 1 (2025-26) | Year 2 (2026-27) | Year 3 (2027-28) |
|-------------|---------------------|-------------------|--------------------|
| Project 1.1 | Initial synthesis | Cell-based assays | Lead optimization |
| Project 1.2 | Theory development | Search algorithms | Final validation |
| Project 2.1 | Simulation studies | Physical platform | Dynamic tasks |
| Project 2.2 | Algorithm design | Integration | Refinement |
| Project 3.1 | Architecture design | Implementation | Evaluation |
| Project 3.2 | Initial experiments | Refinement | Integration |
| Project 4.1 | Dataset preparation | Model development | Real-world testing |
| Project 4.2 | Theory development | Implementation | Validation |

8 Broader impacts

8.1 Workforce Development and Diversity

Ph.D. Student Training and Placement. The PI maintains an active research group of **10 Ph.D. students** at UCLA, with strong representation from underrepresented groups including two Hispanic students (Andrew Lizarraga and Edouardo Honig) and three female students (Lucy Zhao, Yasi Zhang and Andrea Kang). These students are engaged in cutting-edge research in machine learning and will contribute to the proposed research directions. The PI’s mentoring has led to successful academic and industry placements: two recent graduates (Tian Han and Spencer Frei) secured tenure-track faculty positions, while other graduates have joined leading AI research teams - two at Google Deepmind (Ruiqi Gao and Sirui Xie, with Gao being female), two at Salesforce Research (Bo Pang, Erik Nijkamp), and two at Amazon Research (Tianyang Zhao, Yaxuan Zhu), where they work on large language models and advanced AI systems.

Master’s Student Education and Research. The PI has demonstrated a strong commitment to Master’s education, having mentored **over 30 Master’s students since 2019**, including students from underrepresented groups such as Henry Burton (African American). Through UCLA Statistics Department’s Applied Master’s program, established in 2016, the PI has guided **more than 70 Applied Master’s students since 2018**. Notably, over one-third of these Master’s students are female, with several from minority backgrounds. The PI actively integrates Master’s students into research projects and will continue this practice

in the proposed research.

Undergraduate Research Mentoring. The PI currently supervises **6 UCLA undergraduate students** (Eric Jiang, Kelsey Shan, Aditya Bharath, Richard Cao, Yuer Tang, Akshat Tirumalai) on graduate-level research projects. These students engage in sophisticated machine learning research, and several will participate in the proposed research initiatives.

K-12 Outreach and Education. The PI is actively involved in introducing artificial intelligence to pre-college students. In May 2024, he delivered an educational talk at Orange County School of the Arts in Santa Ana, introducing high school students to modern AI concepts and applications. The PI also provided individual research mentorship to high school student Advait Appajodo on large language models during May-June 2024, demonstrating commitment to early STEM education.

Curriculum Innovation. The PI has developed a comprehensive machine learning curriculum at UCLA spanning undergraduate to graduate levels, creating three distinct courses tailored to different student populations. Most recently, he completed a comprehensive 400+ page textbook supporting two machine learning courses (STATS 231 for Ph.D. and Master’s programs and STATS 413 for Applied Master’s program) taught in Fall 2024, ensuring high-quality educational resources for future generations of students.

8.2 Community Engagement

The PI actively shapes the direction of machine learning research through extensive service to the academic community. He serves as area chair for premier conferences including AAAI (2019, 2020, 2024), CVPR 2019, NeurIPS (2021-2024), and ICML (2023-2025). His editorial roles include associate editor positions at *Transactions on Machine Learning Research (TMLR)*, *Journal of the American Statistical Association (JASA)*, and *Journal of Computational and Graphical Statistics (JCGS)*. The PI contributed as a panelist for the highly attended tutorial on latent diffusion models at NeurIPS 2023, and co-organized a workshop on synthetic data at UCLA in March 2023. Since 2020, he has served as an Amazon scholar at Amazon Research, bridging academic research with industry applications.

9 Results from prior NSF support

PI Wu: Generative Modeling with Short Run Computing, DMS 2015577, 07/01/2020–06/30/2024, \$200,000. **Intellectual merit:** This project focuses on developing generative models and the associated learning and inference algorithms based on short-run Markov chain Monte Carlo (MCMC) sampling. The project has produced more than 30 papers published in the top conferences, such as NeurIPS [69, 22, 106, 109, 57, 107], ICML [70, 71, 105, 102], ICLR [20, 65, 58], CVPR [72, 111, 101, 19, 30, 10], AAAI [21], ECCV [66], AISTAT [8], UAI [49], and journals such as PAMI and Neural Computation etc. [99, 100, 103, 98, 31]. **Broader impacts:** The NSF support helps Wu design and teach 3 courses on machine learning, and helps the PI supervise numerous Ph.D., Master’s, and Applied Master’s students. The NSF support also helps the PI to continue professional activities, e.g., serving as an area chair for NeurIPS 2021-2024, ICML 2023-2025.

References

- [1] Charles Beattie, Joel Z Leibo, Denis Teplyashin, Tom Ward, Marcus Wainwright, Heinrich Küttler, Andrew Lefrancq, Simon Green, Víctor Vitoría, Mohamed Sadik, et al. DeepMind Lab. *arXiv preprint arXiv:1612.03801*, 2016.
- [2] Homanga Bharadhwaj, Aviral Kumar, Nicholas Rhinehart, Sergey Levine, Florian Shkurti, and Animesh Garg. Model-based offline planning. *International Conference on Learning Representations*, 2020.
- [3] G Richard Bickerton, Gaia V Paolini, Jérémy Besnard, Sorel Muresan, and Andrew L Hopkins. Quantifying the chemical beauty of drugs. *Nature Chemistry*, 4(2):90–98, 2012.
- [4] David M Blei, Alp Kucukelbir, and Jon D McAuliffe. Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112(518):859–877, 2017.
- [5] Eric Brochu, Vlad M Cora, and Nando De Freitas. A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *arXiv preprint arXiv:1012.2599*, 2010.
- [6] Tom Brown et al. Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33, 2020.
- [7] Joao Carreira, Eric Noland, Chloe Hillier, and Andrew Zisserman. A short note on the Kinetics-700 human action dataset. In *arXiv preprint arXiv:1907.06987*, 2019.
- [8] Cristian I Challu, Peihong Jiang, Ying Nian Wu, and Laurent Callot. Deep generative model with hierarchical latent factors for time series anomaly detection. In *International Conference on Artificial Intelligence and Statistics*, pages 1643–1654. PMLR, 2022.
- [9] Ciprian Chelba, Tomas Mikolov, Mike Schuster, Qi Ge, Thorsten Brants, Phillipp Koehn, and Tony Robinson. One billion word benchmark for measuring progress in statistical language modeling. *arXiv preprint arXiv:1312.3005*, 2013.
- [10] Jiali Cui, Ying Nian Wu, and Tian Han. Learning hierarchical features with joint latent space energy-based prior. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2218–2227, 2023.
- [11] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. Scaling egocentric vision: The EPIC-KITCHENS dataset. In *Proceedings of the European Conference on Computer Vision*, pages 720–736, 2018.
- [12] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34, 2021.
- [13] Peter Eckmann, Kunyang Sun, Bo Zhao, Mudong Feng, Michael K Gilson, and Rose Yu. Limo: Latent inceptionism for targeted molecule generation. *arXiv preprint arXiv:2206.09010*, 2022.
- [14] Peter Ertl and Ansgar Schuffenhauer. Estimation of synthetic accessibility score of drug-like molecules based on molecular complexity and fragment contributions. *Journal of Cheminformatics*, 1(1):1–11, 2009.
- [15] Patrick Esser, Robin Rombach, and Björn Ommer. Taming transformers for high-resolution image synthesis. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12873–12883, 2021.

- [16] Guoyi Fu, Abdelhadi Mohammed, and Xin Wang. Towards efficient multi-objective optimization: Multi-objective lower bounds and reduced equivalent bi-objective problems. *2016 IEEE Congress on Evolutionary Computation*, pages 4651–4658, 2016.
- [17] Justin Fu, Aviral Kumar, Ofir Nachum, George Tucker, and Sergey Levine. D4RL: Datasets for deep data-driven reinforcement learning. *arXiv preprint arXiv:2004.07219*, 2020.
- [18] Scott Fujimoto, David Meger, and Doina Precup. Off-policy deep reinforcement learning without exploration. *International Conference on Machine Learning*, pages 2052–2062, 2019.
- [19] Ruiqi Gao, Erik Nijkamp, Diederik P Kingma, Zhen Xu, Andrew M Dai, and Ying Nian Wu. Flow contrastive estimation of energy-based models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7518–7528, 2020.
- [20] Ruiqi Gao, Yang Song, Ben Poole, Ying Nian Wu, and Diederik P Kingma. Learning energy-based models by diffusion recovery likelihood. In *International Conference on Learning Representations*, 2020.
- [21] Ruiqi Gao, Jianwen Xie, Siyuan Huang, Ren Yufan, Song-Chun Zhu, and Ying Nian Wu. Learning v1 simple cells with vector representations of local contents and matrix representations of local motions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2022.
- [22] Ruiqi Gao, Jianwen Xie, Xue-Xin Wei, Song-Chun Zhu, and Ying Nian Wu. On path integration of grid cells: group representation and isotropic scaling. In *Neural Information Processing Systems*, 2021.
- [23] Andrew Gelman. *Multilevel (hierarchical) modeling: what it can and cannot do*, volume 48. 2006.
- [24] Andrew Gelman, Hal S Stern, John B Carlin, David B Dunson, Aki Vehtari, and Donald B Rubin. *Bayesian data analysis*. Chapman and Hall/CRC, 2013.
- [25] Aaron Gokaslan and Vanya Cohen. OpenWebText corpus, 2019.
- [26] Rafael Gómez-Bombarelli et al. Automatic chemical design using a data-driven continuous representation of molecules. *ACS Central Science*, 4(2):268–276, 2018.
- [27] Abhishek Gupta, Vikash Kumar, Corey Lynch, Sergey Levine, and Karol Hausman. Relay policy learning: Solving long-horizon tasks via imitation and reinforcement learning. *Conference on Robot Learning*, pages 1025–1037, 2019.
- [28] Danijar Hafner et al. Learning latent dynamics for planning from pixels. *International Conference on Machine Learning*, 2019.
- [29] Jesse Michael Han, Stanislas Polu, Jerry Wu, Mislav Balunovic, Kunhao Zheng, and Nate Kushman. Proof artifact co-training for theorem proving with language models. In *International Conference on Learning Representations*, 2022.
- [30] Tian Han, Erik Nijkamp, Linqi Zhou, Bo Pang, Song-Chun Zhu, and Ying Nian Wu. Joint training of variational auto-encoder and latent energy-based model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7978–7987, 2020.
- [31] Tian Han, Xianglei Xing, Jiawen Wu, and Ying Nian Wu. Replicating neuroscience observations on ml/mf and am face patches by deep generative model. *Neural computation*, 31(12):2348–2367, 2019.
- [32] Peter E Hart, Nils J Nilsson, and Bertram Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE transactions on Systems Science and Cybernetics*, 4(2):100–107, 1968.

- [33] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33, 2020.
- [34] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *Advances in Neural Information Processing Systems*, 35:1–13, 2022.
- [35] Matthew D Hoffman, David M Blei, Chong Wang, and John Paisley. Stochastic variational inference. *Journal of Machine Learning Research*, 14(1):1303–1347, 2013.
- [36] Martin Holzer, Frank Schulz, Dorothea Wagner, and Thomas Willhalm. Engineering multilevel overlay graphs for shortest-path queries. *ACM Journal of Experimental Algorithmics*, 13:2–5, 2009.
- [37] Yining Hong, Beide Liu, Maxine Wu, Yuanhao Zhai, Kai-Wei Chang, Lingjie Li, Kevin Lin, Chung-Ching Lin, Jianfeng Wang, Zhengyuan Yang, et al. Slowfast-vgen: Slow-fast learning for action-driven long video generation. *arXiv preprint arXiv:2410.23277*, 2024.
- [38] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.
- [39] Po-Ssu Huang, Scott E Boyken, and David Baker. The coming of age of de novo protein design. *Nature*, 537(7620):320–327, 2016.
- [40] Jason P Hughes, Stephen Rees, S Barry Kalindjian, and Karen L Philpott. Principles of early drug discovery. *British Journal of Pharmacology*, 162(6):1239–1249, 2011.
- [41] Wengong Jin, Regina Barzilay, and Tommi Jaakkola. Junction tree variational autoencoder for molecular graph generation. *International Conference on Machine Learning*, pages 2323–2332, 2018.
- [42] Matthew Johnson, Katja Hofmann, Tim Hutton, and David Bignell. The Malmo platform for artificial intelligence experimentation. *International Joint Conference on Artificial Intelligence*, pages 4246–4247, 2016.
- [43] Michael I Jordan, Zoubin Ghahramani, Tommi S Jaakkola, and Lawrence K Saul. An introduction to variational methods for graphical models. *Machine Learning*, 37(2):183–233, 1999.
- [44] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.
- [45] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- [46] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.
- [47] Deqian Kong, Yuhao Huang, Jianwen Xie, Edouardo Honig, Ming Xu, Shuanghong Xue, Pei Lin, Sanping Zhou, Sheng Zhong, Nanning Zheng, and Ying Nian Wu. Molecule design by latent prompt transformer. In *Advances in Neural Information Processing Systems*, 2024.
- [48] Deqian Kong, Bo Pang, Tian Han, and Ying Nian Wu. Molecule design by latent space energy-based modeling and gradual distribution shifting. In *Conference on Uncertainty in Artificial Intelligence (UAI)*, volume 216, pages 1109–1120, 2023.
- [49] Deqian Kong, Bo Pang, Tian Han, and Ying Nian Wu. Molecule design by latent space energy-based modeling and gradual distribution shifting. In *Uncertainty in Artificial Intelligence*, pages 1109–1120. PMLR, 2023.

- [50] Deqian Kong, Dehong Xu, Minglu Zhao, Bo Pang, Jianwen Xie, Andrew Lizarraga, Yuhao Huang, Sirui Xie, and Ying Nian Wu. Latent plan transformer for trajectory abstraction: Planning as latent space inference. In *Advances in Neural Information Processing Systems*, 2024.
- [51] Mario Krenn, Florian Häse, AkshatKumar Nigam, Pascal Friederich, and Alan Aspuru-Guzik. Self-referencing embedded strings (selfies): A 100% robust molecular string representation. *Machine Learning: Science and Technology*, 1(4):045024, 2020.
- [52] Dharshan Kumaran, Demis Hassabis, and James L McClelland. What learning systems do intelligent agents need? complementary learning systems theory updated. *Trends in cognitive sciences*, 20(7):512–534, 2016.
- [53] Greg Landrum et al. RDKit: Open-source cheminformatics. URL <http://www.rdkit.org>, 2006.
- [54] Yann LeCun, Sumit Chopra, Raia Hadsell, M Ranzato, and F Huang. A tutorial on energy-based learning. *Predicting structured data*, 1(0), 2006.
- [55] Sergey Levine, Aviral Kumar, George Tucker, and Justin Fu. Offline reinforcement learning: Tutorial, review, and perspectives on open problems. *arXiv preprint arXiv:2005.01643*, 2020.
- [56] Xiang Lisa Li, John Thickstun, Ishaan Gulrajani, Percy Liang, and Tatsunori B Hashimoto. Diffusion-LM improves controllable text generation. *Advances in Neural Information Processing Systems*, 35:1–13, 2022.
- [57] Pan Lu, Baolin Peng, Hao Cheng, Michel Galley, Kai-Wei Chang, Ying Nian Wu, Song-Chun Zhu, and Jianfeng Gao. Chameleon: Plug-and-play compositional reasoning with large language models. *arXiv preprint arXiv:2304.09842*, 2023.
- [58] Pan Lu, Liang Qiu, Kai-Wei Chang, Ying Nian Wu, Song-Chun Zhu, Tanmay Rajpurohit, Peter Clark, and Ashwin Kalyan. Dynamic prompt learning via policy gradient for semi-structured mathematical reasoning. *arXiv preprint arXiv:2209.14610*, 2022.
- [59] Mitchell P Marcus, Mary Ann Marcinkiewicz, and Beatrice Santorini. Building a large annotated corpus of English: The Penn Treebank. *Computational linguistics*, 19(2):313–330, 1993.
- [60] Claire Marks, Tianqi Shi, and Charlotte M Deane. Protein design by deep network hallucination. *Nature Machine Intelligence*, 4(5):462–470, 2022.
- [61] James L McClelland, Bruce L McNaughton, and Randall C O’Reilly. Why there are complementary learning systems in the hippocampus and neocortex: insights from the successes and failures of connectionist models of learning and memory. *Psychological Review*, 102(3):419, 1995.
- [62] Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. *arXiv preprint arXiv:1609.07843*, 2016.
- [63] Edouard Mullarky, Natasha C Lucki, Reza Beheshti Zavareh, Justin L Anglin, Ana P Gomes, Brandon N Nicolay, Jenny C Y Wong, Stefan Christen, Hidenori Takahashi, Param Priya Singh, et al. Targeting PHGDH in cancer: challenges and opportunities. *Molecular Cell*, 76(6):845–857, 2019.
- [64] Edouard Mullarky, Katherine R Mattaini, Matthew G Vander Heiden, Lewis C Cantley, and Jason W Locasale. PHGDH amplification and altered glucose metabolism in human melanoma. *Pigment Cell & Melanoma Research*, 29(4):441–447, 2016.
- [65] Erik Nijkamp, Ruiqi Gao, Pavel Sountsov, Srinivas Vasudevan, Bo Pang, Song-Chun Zhu, and Ying Nian Wu. Mcmc should mix: Learning energy-based model with neural transport latent space mcmc. In *International Conference on Learning Representations*, 2021.

- [66] Erik Nijkamp, Bo Pang, Tian Han, Linqi Zhou, Song-Chun Zhu, and Ying Nian Wu. Learning multi-layer latent variable model via variational optimization of short run mcmc for approximate inference. In *European Conference on Computer Vision*, pages 361–378. Springer, 2020.
- [67] John O’Keefe and Lynn Nadel. *The hippocampus as a cognitive map*. Oxford: Clarendon Press, 1978.
- [68] Michael E Pacold, Kyle R Brimacombe, Sze Ham Chan, Jason M Rohde, Caroline A Lewis, Lotte JYM Swier, Richard Possemato, Walter W Chen, Lucas B Sullivan, Brian P Fiske, et al. A PHGDH inhibitor reveals coordination of serine synthesis and one-carbon unit fate. *Nature Chemical Biology*, 12(6):452–458, 2016.
- [69] Bo Pang, Tian Han, Erik Nijkamp, Song-Chun Zhu, and Ying Nian Wu. Learning latent space energy-based prior model. *Advances in Neural Information Processing Systems*, 33, 2020.
- [70] Bo Pang and Ying Nian Wu. Latent space energy-based model of symbol-vector coupling for text generation and classification. In *International Conference on Machine Learning*, pages 8359–8370. PMLR, 2021.
- [71] Bo Pang and Ying Nian Wu. Latent space energy-based model of symbol-vector coupling for text generation and classification. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8359–8370. PMLR, 18–24 Jul 2021.
- [72] Bo Pang, Tianyang Zhao, Xu Xie, and Ying Nian Wu. Trajectory prediction with latent belief energy-based model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11814–11824, June 2021.
- [73] Denis Paperno, Germán Kruszewski, Angeliki Lazaridou, Ngoc Quan Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel Fernández. The LAMBADA dataset: Word prediction requiring a broad discourse context. *arXiv preprint arXiv:1606.06031*, 2016.
- [74] Andrew Parker. In the blink of an eye: how vision sparked the big bang of evolution. *Basic Books*, 2003.
- [75] Stefania Perspicace, Arne C Rufer, Ralf Thoma, Sebastian Krapp, et al. Surface plasmon resonance-based screening procedure for identification of small molecules interacting with hepatitis C virus RNA. *FEBS open bio*, 3:444–455, 2013.
- [76] Stanislas Polu, Jesse Michael Han, Kunhao Zheng, Mantas Baksys, Igor Babuschkin, and Richard Sutton. Formal mathematics statement curriculum learning. *arXiv preprint arXiv:2202.01344*, 2022.
- [77] Richard Possemato, Kevin M Marks, Yoav D Shaul, Michael E Pacold, Dohoon Kim, Kıvanç Birsoy, Shalini Sethumadhavan, Hee-Kwon Woo, Hyun G Jang, Abhishek K Jha, et al. Functional genomics reveal that the serine synthesis pathway is essential in breast cancer. *Nature*, 476(7360):346–350, 2011.
- [78] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.
- [79] Carl Edward Rasmussen. *Gaussian processes in machine learning*. MIT press, 2003.
- [80] Michelle A Reid, Andrew E Allen, Shuang Liu, Marco V Liberti, Pengtao Liu, Xiaojing Liu, Zheng Dai, Xiang Gao, Qiu Wang, Yang Liu, et al. Development of a novel series of 3-PHGDH inhibitors. *Bioorganic & Medicinal Chemistry*, 26(11):2965–2972, 2018.

- [81] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022.
- [82] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015*, pages 234–241, 2015.
- [83] David Santos-Martins, Jerome Eberhardt, Giulia Bianco, Leonardo Solis-Vasquez, Forli Ambrosio, Andreas Koch, and Stefano Forli. Accelerating AutoDock4 with GPUs and gradient-based local search. *Journal of Chemical Information and Modeling*, 61(8):3852–3866, 2021.
- [84] Burr Settles. Active learning literature survey. *University of Wisconsin-Madison Department of Computer Sciences*, 2009.
- [85] David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dhharshan Kumaran, Thore Graepel, et al. A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. *Science*, 362(6419):1140–1144, 2018.
- [86] Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical bayesian optimization of machine learning algorithms. *Advances in Neural Information Processing Systems*, 25:2951–2959, 2012.
- [87] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. *International Conference on Machine Learning*, pages 2256–2265, 2015.
- [88] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *International Conference on Learning Representations*, 2020.
- [89] Richard S Sutton and Andrew G Barto. *Reinforcement Learning: An Introduction*. MIT Press, 2 edition, 2018.
- [90] Romal Thoppilan, Daniel De Freitas, Jamie Hall, Noam Shazeer, Apoorv Kulshreshtha, Heng-Tze Cheng, Alicia Jin, Taylor Bos, Leslie Baker, Yu Du, et al. LaMDA: Language models for dialog applications. *arXiv preprint arXiv:2201.08239*, 2022.
- [91] Naftali Tishby, Fernando C Pereira, and William Bialek. The information bottleneck method. *arXiv preprint physics/0004057*, 2000.
- [92] Emanuel Todorov, Tom Erez, and Yuval Tassa. MuJoCo: A physics engine for model-based control. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5026–5033, 2012.
- [93] Edward C Tolman. Cognitive maps in rats and men. *Psychological review*, 55(4):189, 1948.
- [94] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, 30:5998–6008, 2017.
- [95] Qiu Wang, Marco V Liberti, Pengtao Liu, Xianrui Deng, Yang Liu, Jason W Locasale, and Luhua Lai. Structural basis for the inhibition of human 3-PHGDH by EBI-9007856. *Chemical Science*, 10(9):2725–2731, 2019.
- [96] David Weininger. Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. *Journal of chemical information and computer sciences*, 28(1):31–36, 1988.

- [97] Fei Xia, Amir R Zamir, Zhiyang He, Alexander Sax, Jitendra Malik, and Silvio Savarese. Gibson env: Real-world perception for embodied agents. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9068–9079, 2018.
- [98] Jianwen Xie, Ruiqi Gao, Erik Nijkamp, Song-Chun Zhu, and Ying Nian Wu. Representation learning: A statistical perspective. *Annual Review of Statistics and Its Application*, 7:303–335, 2020.
- [99] Jianwen Xie, Zilong Zheng, Xiaolin Fang, Song-Chun Zhu, and Ying Nian Wu. Cooperative training of fast thinking initializer and slow thinking solver for conditional learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.
- [100] Xianglei Xing, Ruiqi Gao, Tian Han, Song-Chun Zhu, and Ying Nian Wu. Deformable generator networks: unsupervised disentanglement of appearance and geometry. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.
- [101] Xianglei Xing, Tianfu Wu, Song-Chun Zhu, and Ying Nian Wu. Inducing hierarchical compositional model by sparsifying generator network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14296–14305, 2020.
- [102] Yan Xu, Deqian Kong, Dehong Xu, Ziwei Ji, Bo Pang, Pascale Fung, and Ying Nian Wu. Diverse and faithful knowledge-grounded dialogue generation via sequential posterior inference. *arXiv preprint arXiv:2306.01153*, 2023.
- [103] Yifei Xu, Jianwen Xie, Tianyang Zhao, Chris Baker, Yibiao Zhao, and Ying Nian Wu. Energy-based continuous inverse optimal control. *IEEE Transactions on Neural Networks and Learning Systems*, 2022.
- [104] Jiaxuan You, Bowen Liu, Zhitao Ying, Vijay Pande, and Jure Leskovec. Graph convolutional policy network for goal-directed molecular graph generation. *Advances in Neural Information Processing Systems*, 31:6410–6421, 2018.
- [105] Peiyu Yu, Sirui Xie, Xiaojian Ma, Baoxiong Jia, Bo Pang, Ruiqi Gao, Yixin Zhu, Song-Chun Zhu, and Ying Nian Wu. Latent diffusion energy-based model for interpretable text modeling. *arXiv preprint arXiv:2206.05895*, 2022.
- [106] Peiyu Yu, Sirui Xie, Xiaojian Ma, Yixin Zhu, Ying Nian Wu, and Song-Chun Zhu. Unsupervised foreground extraction via deep region competition. *Advances in Neural Information Processing Systems*, 34:14264–14279, 2021.
- [107] Peiyu Yu, Yaxuan Zhu, Sirui Xie, Xiaojian Shawn Ma, Ruiqi Gao, Song-Chun Zhu, and Ying Nian Wu. Learning energy-based prior model with diffusion-amortized mcmc. *Advances in Neural Information Processing Systems*, 36:42717–42747, 2023.
- [108] Zheng Yu, Xiang Zhang, Yichao Zhu, and Boxing Chen. Diffusion models for non-autoregressive text generation. *arXiv preprint arXiv:2212.09749*, 2022.
- [109] Wenhao Zhang, Ying Nian Wu, and Si Wu. Translation-equivariant representation in recurrent networks with a continuous manifold of attractors. *Advances in Neural Information Processing Systems*, 35:15770–15783, 2022.
- [110] Zhenpeng Zhou, Steven Kearnes, Li Li, Richard N Zare, and Patrick Riley. Optimization of molecules via deep reinforcement learning. *Scientific Reports*, 9(1):1–10, 2019.
- [111] Yaxuan Zhu, Ruiqi Gao, Siyuan Huang, Song-Chun Zhu, and Ying Nian Wu. Learning neural representation of camera pose with matrix representation of pose shift via view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9959–9968, 2021.