Cognitively Inspired Energy-Based World Models

A Technical Report submitted to the Department of Computer Science

Presented to the Faculty of the School of Engineering and Applied Science

University of Virginia • Charlottesville, Virginia

In Partial Fulfillment of the Requirements for the Degree

Bachelor of Science, School of Engineering

Ganesh Nanduru

Spring, 2024

Technical Project Team Members
Alexi Gladstone
Md Mofijul Islam
Aman Chadha

On my honor as a University Student, I have neither given nor received unauthorized aid on this assignment as defined by the Honor Guidelines for Thesis-Related Assignments

Advisor

Jundong Li, Department of Computer Science

Cognitively Inspired Energy-Based World Models

Alexi Gladstone[‡*], **Ganesh Nanduru**[‡], **Md Mofijul Islam**[‡§†]
Aman Chadha[•§†], **Jundong Li**[‡], **Tariq Iqbal**[‡]
[‡]University of Virginia, [•]Stanford University, [§]Amazon GenAI

Abstract

One of the predominant methods for training world models is autoregressive prediction in the output space of the next element of a sequence. In Natural Language Processing (NLP), this takes the form of Large Language Models (LLMs) predicting the next token; in Computer Vision (CV), this takes the form of autoregressive models predicting the next frame/token/pixel. However, this approach differs from human cognition in several respects. First, human predictions about the future actively influence internal cognitive processes. Second, humans naturally evaluate the plausibility of predictions regarding future states. Based on this capability, and third, by assessing when predictions are sufficient, humans allocate a dynamic amount of time to make a prediction. This adaptive process is analogous to System 2 thinking in psychology. All these capabilities are fundamental to the success of humans at high-level reasoning and planning. Therefore, to address the limitations of traditional autoregressive models lacking these human-like capabilities, we introduce Energy-Based World Models (EBWM). EBWM involves training an Energy-Based Model (EBM) to predict the compatibility of a given context and a predicted future state. In doing so, EBWM enables models to achieve all three facets of human cognition described. Moreover, we developed a variant of the traditional autoregressive transformer tailored for Energy-Based models, termed the Energy-Based Transformer (EBT). Our results demonstrate that EBWM scales better with data and GPU Hours than traditional autoregressive transformers in CV, and that EBWM offers promising early scaling in NLP. Consequently, this approach offers an exciting path toward training future models capable of System 2 thinking and intelligently searching across state spaces.

1 Introduction

Self-Supervised Learning (SSL) has established itself as a powerful method for training large foundation models in Computer Vision (CV) [6–11], Natural Language Processing (NLP) [12–17], and audio/speech processing [18–20]. Owing to several of these foundation models having a general understanding of a variety of concepts, these models have found numerous use cases. For example, Large Language Models (LLMs) have been used to augment productivity [21, 22], video generation models have been used to generate realistic videos [23], and audio/speech models have been used for conversational AI [24, 25].

Within SSL, one common training approach for achieving such general representations has been autoregressive models predicting the next element of a sequence in the output space. In NLP, this takes the form of predicting the next token [26], in CV, models are trained to predict the next frame [27, 28], next frame's feature [29] or a discretized token (i.e. via Vector Quantization [30–35]), and in Audio Processing, a commonly used approach involves predicting discretized speech units [36, 19, 37] or continuous waveforms [18]. It has even been shown that some approaches can be generalized across modalities [38]. These autoregressive models, which predict the next element of a

[*]Correspondence to Alexi Gladstone: alexigladstone@gmail.com, https://alexiglad.github.io/.
[†]Work does not relate to position at Amazon.

Preprint. Under review.

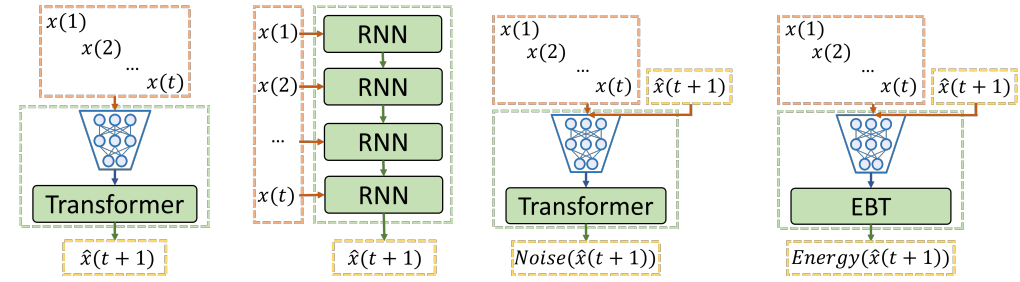


Figure 1: Comparison of traditional autoregressive approaches and EBWM. (a) Autoregressive (AR) Transformer is the most common, and what we compare scaling results to in this paper, with (b) RNN becoming more popular within the past year [1, 2]. (c) is the most similar to EBWM, being able to allocate a dynamic amount of computation during inference, but predicts the noise rather than the energy [3]. Several architectures are interchangeable, such as RetNet for Transformers [4], or Mamba/LSTM [1, 5] for RNNs. EBWM, being an Energy-Based Model (EBM), has all four cognitive capabilities described. Trans. stands for Transformer.

sequence in the output space, can be broadly classified as world models (more on this definition is in Section 2), and more specifically, as Traditional Autoregressive Models (TAMs). These are shown in Fig. 1(a) and Fig. 1(b).

World models compared to human cognition: World models have achieved remarkable feats in recent years, scoring higher than the average human on the SAT and LSAT [12] as well as being able to generate synthetic photo-realistic videos of hypothetical scenarios [39]. Yet, these models still struggle to perform seemingly simple and innate human capabilities, such as reasoning, planning, or thinking about problems over an extended horizon of time [40–44]. This phenomenon is broadly known as Moravec's Paradox [45]. Along similar lines, recent work [46] has shown that many generative AI models are better at generating than they are at understanding, achieving expert performance on tasks such as image generation but failing to answer simple questions about images that a preschooler could answer. This also contrasts from humans, who generally develop the skill of understanding before generating [46, 47]. Thus, it's possible a significant portion of the disparity between modern AI and human cognition is caused by fundamental differences in the way TAMs work as compared to the human brain. Achieving human-like cognition in models could enhance their ability to reason, plan, and perform System 2 thinking at the same level as humans. As such, in this work, we design an architecture to achieve the following four core cognitive capabilities believed to be fundamental to high-level reasoning and planning in humans:

Facet (1): Predictions shape internal state: It is widely supported that when humans predict the future, these thoughts naturally shape our brains' internal state [48–51]. However, TAMs, having predictions made in the output space, do not have their internal state shaped by a given prediction (see Fig. 1). [3]

Facet (2): Evaluation of predictions: There is strong evidence supporting the idea that humans naturally evaluate the plausibility of predictions [52–54]. TAMs, making predictions of future states in the output space, cannot evaluate the strength or plausibility of predictions.

Facet (3): Dynamic allocation of resources: The idea that humans naturally dedicate various amounts of time towards making predictions or reasoning is widely supported in psychology and neuroscience [55–57]. As the difficulty of tasks humans face varies widely, the ability to adjust the magnitude of computational resources allocated towards a task is fundamental to success.

Facet (4): Modeling uncertainty in continuous state spaces: LLMs can naturally model uncertainty through the allocation of probability mass across token classes [58]. In the context of continuous state

[3] While one could argue that re-incorporating these predictions within the context of models can alter the model's internal state, it is crucial to note that in the realm of TAMs, such predictions are treated as ground truth (through teacher forcing during training). Consequently, these models are incapable of discerning between actual observations and their own predictions.

Table 1: All four architectures shown in Fig. 1 and whether or not they have the four cognitive facets described in Section 1. The transformer architecture refers to standard autoregressive transformers. We classify Traditional Transformers and RNNs as TAMs. *Diffusion models are not generally deemed architectures but rather an approach. Despite being less common than autoregressive approaches, we wanted to include them since they have more facets than TAMs.

Architecture	Predictions affect internal state 1	Evaluation of predictions 1	Dynamic compute allocation 1	Uncertainty modeling 1
Transformers	✗	✗	✗	✗
RNN Variants	✗	✗	✗	✗
Diffusion*	✓	✗	✓	✗
EBWM	✓	✓	✓	✓

spaces, such as in CV, without the usage of discretization schemes this is not possible with TAMs. This capability is particularly important in situations where uncertainty is essential, and is a natural capability of humans [59–61].

To achieve all four cognitive facets described, we approach the problem of world modeling as *making future state predictions in the input space* and predicting the *energy/compatibility* of these future state predictions with the current context through the usage of an Energy-Based Model (EBM). Making predictions in the input space achieves cognitive facets (1) and (3), while predicting the compatibility of future state predictions with the current context achieves cognitive facets (2) and (4) (shown in Table 1). The architecture of EBWM is shown in Fig 1(d).

Additionally, as EBMs have struggled to compete with models using modern architectures, we design a domain-agnostic EBM transformer architecture named the Energy-Based Transformer (EBT). We will move the whole EBM paradigm forward by releasing the implementation of EBT, which allows for the parallelization of multiple predictions at once, similar to traditional transformers [62]. Our experiments demonstrate the scalability of EBT, achieving better scaling in terms of data and GPU hours in CV when compared to TAMs on different datasets.

The key contributions of our work can be summarized as follows:

- We propose a new architecture that is inspired by four core facets of human cognition for training world models, EBWM, and lay the ground work for the techniques necessary to train and implement EBWM across domains. We do extensive analyses regarding different design choices to further support EBWM's design.
- We design and experiment with a domain-agnostic variant of the transformer architecture, EBT, specifically for EBMs.
- We implement EBWM for training world models in CV and NLP. Our findings indicate that EBWM scales similarly to traditional autoregressive transformers, despite being qualitatively different in allowing for the four facets of human cognition described.

2 Background: World Model primer

Autoregressive models predicting the next element of a sequence can be broadly classified as a specific type of world model. Below, we justify this through a precise definition, while giving some additional background to clarify synonymic terms.

World models have been broadly referred to as "internal models of how the world works" [63] and "predictive model of the future that learns a general representation of the world" [33]. More precisely, we define world models most broadly as models that given a state, a previous estimate of the state, a latent variable, and an action, predict the next state [64, 33, 65, 63]. This idea can be formalized as the following ([64]):

$$s(t+1) = F(s(t), x(t), a(t), z(t)), \tag{1}$$

where $x(t)$ is an observation, $s(t)$ is a previous estimate of the world, $a(t)$ is an action, $z(t)$ is a latent variable, and F is the function utilized [64]. The previous estimates of the world are often

encapsulated within past observations. Additionally, most world models used in reinforcement learning are not conditioned on a latent variable, and therefore take the following form:

$$s(t+1) = F(x(1), x(2), ..., x(t), a(t)). \tag{2}$$

Therefore, traditional autoregressive models that have been used in NLP (LLMs), vision (video generation models), and audio processing (speech generation models), can be seen as a specific type of world model where the state is defined by the observations and no action or latent variable is being conditioned on [66, 64]. These types of world models simplify to the following mathematically, where $(x(1), ..., x(t))$ is often referred to as the context:

$$x(t+1) = F(x(1), x(2), ..., x(t)) \tag{3}$$

Furthermore, it's worth noting that the idea of world models has a backing in both robotics and in neuroscience. World models are analogous to Model Predictive Control in robotics [67, 63]. Similarly, world models, when the observations are sensory information and the state is approximated using sensory information, are comparable to predictive coding [51] from neuroscience. The primary difference lies in the application of these terms [68]. In this work we use the term world model to broadly refer to the most general world model definition, Model Predictive Control, and predictive coding.

3 Related Work

3.1 Traditional autoregressive world models

Several world models across domains have been proposed in the literature, including LLMs [17, 69, 70, 66], video world models [71–73], and autoregressive audio processing world models [20, 18, 74]. Similarly, in Reinforcement Learning, world models conditioned on an action are common [75–77, 33].

Transformers and Variants: Several world models [26, 17, 78] have used the transformer architecture [62], which has become ubiquitous in sequence modeling due to their efficient usage of computation. In the context of traditional autoregressive transformers making predictions of the next state in the output space, as in Fig. 1(a), these models also lack several of the cognitive facets discussed in Section 1 (Table 1). This includes the ability to leverage a dynamic amount of computation for making predictions or the ability to evaluate the plausibility of predictions.

RNNs and Variants: Recently, several RNN variants (shown in Fig. 1(b)) have emerged to reduce memory bottlenecks and achieve faster inference [1, 2]. These approaches have scaled similarly to transformers in autoregressive sequence modeling, and achieve better memory efficiency and reduced latency. However, traditional RNNs making predictions of the next state in the output space also lack several of the cognitive facets discussed in Section 1 (see Table 1).

3.2 Autoregressive world models with dynamic computation

3.2.1 Diffusion

Several existing works have attempted to enable pre-trained autoregressive models to leverage extra computation during inference to make high level decisions or solve challenging problems. The most common instance of this is diffusion models (Fig. 1(c)), where using multiple forward passes for generating a prediction is a core aspect of both training and inference [3, 79]. Although this idea was originally developed in CV, it has recently been popularized in audio processing [18] as well as NLP [80]. These models also lack two of the cognitive facets discussed in Section 1 (Table 1), through not having the ability to evaluate predictions or model uncertainty in continuous state spaces.

3.2.2 Dynamic computation in LLMs

The ability to leverage a dynamic amount of computation is closely mirrored in chain-of-thought prompting, where an LLM is prompted to elicit its thought process before giving an answer [81].

This gives an LLM additional computational depth based off the number of tokens decoded before making a prediction, and as a result significantly improves performance in many cases. Similarly, a recently developed solution via the concept of a thinking token [82] enabled LLMs to leverage multiple forward passes in predicting the next token. However, both these approaches do not achieve all four cognitive facets described in Section 1 (see Section E for more details).

3.3 Energy-Based Models

One contribution of this work was the design of a custom architecture for EBM's called the Energy-Based Transformer (EBT). Somewhat similar is the work of the Energy Transformer [83]. Despite strong similarity in the names of these architectures, however, they are very different–with the primary similarity in architectures being the usage of attention mechanisms as well as a global energy function. The existing work integrated ideas from Modern Hopfield Networks, including RNNs, whereas in our work the architecture is non-recurrent and does not use associative memories. Additionally, EBT differs with its focus on autogressive modeling, which this previous work did not experiment with.

The most similar works to ours involve autoregressive Energy-Based Models, including E-ARM [84], EBR [85], and Residual EBMs [86]. E-ARM involves adding an objective to the learning process to turn a traditional autoregressive model into an EBM, and as such does not achieve three of the cognitive characteristics discussed in Section 1. EBR and Residual EBMs involve the training of an EBM on top of an already pre-trained autoregressive language model. Both works, however, leverage a contrastive objective, which suffers from the curse of dimensionality. EBWM differs from these works through leveraging a reconstruction objective (more details in Section 5.1), being domain agnostic, and by not requiring a pre-trained model to work on top of (EBWM is a standalone solution). We also develop and leverage the EBT to achieve better scaling–previous works did not use such modern architectures.

4 Energy-Based World Models (EBWM) Intuition

Energy-Based Models (EBMs) are a class of models that associate a scalar energy value with each configuration of input variables, producing lower values for highly compatible inputs and higher values for less compatible inputs [63]. EBWM leverages this, and is trained to predict *how compatible* a given context and predicted future state are, where high energy corresponds to low compatibility and low energy corresponds to high compatibility.

Intelligent search: Recent work has revealed a characteristic of world models, deemed "The Generative AI Paradox," where these models have achieved superhuman generative capabilities, being able to outperform even the best humans, but paradoxically lack the ability to discriminate or "understand" states at the level of an average human [46] or reason at the level of an infant [87]. This is problematic as the ability to discriminate is essential in determining what states are plausible and desired when reasoning and planning. For example, in the realm of NLP, researchers have explored approaches such as tree search for generating several plausible answers [88, 89]. One limitation of this approach that has often been stated is that LLMs know how to generate the right answer, but they do not know how to *choose* this answer [90]. This has made tree search extremely computationally expensive, as thousands or even millions of samples need to be generated to achieve optimal performance [88]. Therefore, world models with the ability to directly evaluate the plausibility of a proposed state (1) offer a promising path in solving this challenge of searching the state space intelligently. EBWM, being an EBM, can use its scalar energy value to directly evaluate a state and MCMC (more details on MCMC can be found in Section 5.2) to directly improve upon a state–thereby offering an opportunity to both select states as well as efficiently search the state space. This scalar energy value also offers benefits in determining when to finish generating, as a cutoff threshold for the energy value can be used to determine when the model has achieved sufficient certainty about a prediction.

System 2 thinking: Consider the task of a person making the most important business decision of their life. Rather than making this decision based on a single fleeting thought, analogous to System 1 thinking from psychology, it's highly probable this person would use System 2 thinking [55], and ponder over a long period of time. This ability for humans to think over an indefinite period of time is believed to be essential for reasoning [55, 91, 92]. TAMs, using a single forward pass to make a prediction of the next state, cannot use a dynamic amount of computation for a prediction

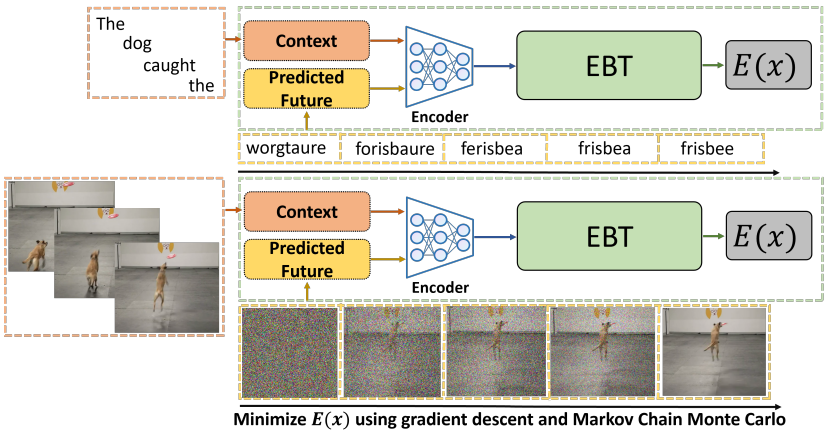


Figure 2: Architecture of EBWM trained models for natural language processing and computer vision. MCMC is used to iteratively refine on the prediction of a future state until convergence of the predicted energy. Each yellow box corresponds to a different "predicted future state" based off of the current MCMC iteration (the first condition in this case is random).

of the future state, making it challenging to reason (see Section E for common counterarguments to this perspective). Diffusion models, in allowing for a dynamic amount of inference cycles when denoising, do allow for dynamic resource allocation (Section 1), but are still challenged by the fact that they cannot evaluate the plausibility of future state predictions 1. EBWM, achieving all four cognitive facets described, can use a dynamic amount of computation and evaluate its predictions, offering a promising path towards System 2 thinking.

Modeling Uncertainty: Consider the case of a vision world model being rolled out to decide whether the left or right fork in a road will be taken. With TAMs, only a single next frame/token/embedding can be predicted as predictions are made in the output space as shown in Fig. 1. Consequently, TAMs would not be able to model the left and right fork in this situation. In cases like these, the modeling of different potential futures is important, and one reason LLMs have succeeded (as states are discrete in LLMs and take the form of tokens). EBWM, being an EBM, can naturally evaluate the energy of different predicted futures, allowing for the modeling of uncertainty.

5 Energy-Based World Models (EBWM) Approach

5.1 Energy-based Models

In traditional approaches to training Energy-Based Models (EBMs), two primary methods have been prevalent: contrastive methods and regularized methods. Contrastive learning approaches [63], such as the most commonly used EBM training approach of contrastive divergence [93, 94], succumb to the curse of dimensionality. Although regularized methods avoid this flaw, they struggle with imposing non-restrictive regularization or inductive biases such as bottlenecks [63].

A recent study demonstrated a regularized approach with weak restrictions [95], incorporating a reconstruction objective. This approach involved training an EBM to create representations such that minimizing the energy predicts a denoised image. Therefore, rather than using a contrastive loss, this approach involved a reconstruction loss calculated in the input space. For continuous state spaces, such as visual signals, this takes the form of a SmoothL1 loss calculated between predicted and ground truth embeddings with $\beta = 1$; and for discrete state spaces, such as in NLP, categorical cross-entropy is used. To our knowledge, this loss has not been used within existing autoregressive EBMs, and aligns with the ideals of training a world model to predict the next state conditioned on all previous observations. We find this simple objective consistently achieves the most stable runs and best loss when compared to other losses (more details on other losses in the supplementary section C.1). As stated in [95], this can intuitively be seen as "folding the encoder-decoder architecture into the forward and backward passes of a model." Detailed pseudocode for the implementation of this approach is in [95] Section A.3.

Table 2: **Ablations on key EBWM design choices in CV.** Convergent refers to whether the reconstruction loss converged. Stable training means that there were no training loss spikes. The model achieved a similar loss with and without Langevin dynamics, so we elected to use the simpler alternative–without Langevin dynamics). Energy Loss and Bounds Loss are described in more detail in Section C.1. *In this case there were no loss spikes and the loss converged but only because the encoder experienced mode collapse.

Design Choice	Stable Training	Convergent
Energy Loss	✗	✓
Bounds Loss	✗	✓
Unfrozen Encoder*	✗	✗
Unclamped MCMC Gradient	✗	✗
Non-Learnable α (MCMC Step Size)	✗	✓
Lower Initial MCMC Step Size	✗	✓
Langevin Dynamics	✓	✓
Learnable Langevin Dynamics	✓	✓
All Specified Design Choices	✓	✓

5.2 Markov Chain Monte Carlo (MCMC) Approaches

One useful property of EBWM is the ability to leverage multiple forward passes in making a prediction. This is done through Markov Chain Monte Carlo (MCMC) methods, where an initial condition is passed into the model and the gradient of this condition with respect to the outputted energy is calculated (as the entire model is differentiable). Using this gradient, the input is updated. This is done iteratively until the convergence of the outputted energy (shown in Fig. 2). The usage of MCMC introduces a couple of new hyperparameters. First is the step size α, which is used to determine how much to multiply the gradient by when adjusting the predicted input. We make this a learnable parameter, but still find it important to tweak the initial value during training as it affects stability (Table 2). Two other hyperparameters introduced by MCMC are the learning rate multiplier of α as well as the number of MCMC steps during training (this can be changed during inference). The values for these hyperparameters are detailed further in Section D.

5.3 Energy-Based Transformer (EBT)

Two of the core mechanisms behind the success of the transformer are its attention mechanism and its parallelizability [62]. In the context of EBMs, the traditional transformer implementation poses a challenge due to the approximation of a joint distribution rather than a conditional distribution. To demonstrate why this poses a challenge, consider the case of the TAMs $n \times n$ attention scores matrix after the causal mask has been applied:

$$\text{scores} = \begin{bmatrix} \alpha_{z_1,z_1} & 0 & \dots & 0 \\ \alpha_{z_2,z_1} & \alpha_{z_2,z_2} & \dots & 0 \\ \dots & \dots & \dots & \dots \\ \alpha_{z_n,z_1} & \alpha_{z_n,z_2} & \dots & \alpha_{z_n,z_n} \end{bmatrix},$$

where α_{z_i,z_j} represents the attention score (probability mass) from state z_i to state z_j. Now, in the case of an EBM, where predictions of future states are made in the input space, the intended $n \times n+1$ attention scores matrix would look like the following:

$$\text{scores} = \begin{bmatrix} \alpha_{z_1,z_1} & \alpha_{z_1,\hat{z}_2} & 0 & \dots & 0 \\ \alpha_{z_2,z_1} & \alpha_{z_2,z_2} & \alpha_{z_2,\hat{z}_3} & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ \alpha_{z_n,z_1} & \alpha_{z_n,z_2} & \alpha_{z_3,z_3} & \dots & \alpha_{z_n,\hat{z}_{n+1}} \end{bmatrix}. \tag{4}$$

This is challenging to compute because each $\hat{z}_i$ along the superdiagonal is unique for its row. Consequently, this matrix cannot be computed with a matrix multiplication (softmax $\left(\frac{QK^T}{\sqrt{d_k}}\right)$) as in regular attention, as every value on the superdiagonal is a prediction and not a past state.

Additionally, in a traditional transformer, if the context length is n, the size of the passed in tensor will be $bs \times n \times d$ where bs is the batch size and d is the embedding dimension. However, since EBMs learn a joint probability distribution, and thus make predictions in the input space, the input tensor needs to be different to allow for inputting future predictions. Therefore, for a context length of n, we define the first $n-1$ elements as z_o, or the original sequence representations, and the final $n-1$ elements as z_p, or the predicted sequence representations.

The values for z_{o1}^n, or the given states, are computed the same as in the original transformer [62], as the attention scores of the original states do not depend on the predicted states. This can be formalized as the following:

$$\text{Attention}(Q_o, K_o, V_o)_{z_{o1}^n} = \text{softmax}\left(\frac{Q_o K_o^T}{\sqrt{d_k}}\right) V_0, \tag{5}$$

Where Q_o, K_o, and V_o are the Query, Key, and Value matrices of the past states z_{o1}^n. Every block of the transformer, the representations of all past states are updated in this manner, independent of the representations of the predicted future states.

For the computation of representations over future states, 3 matrices are also computed, but for the representations of predicted future states rather than past states. We call these Q_p, K_p, and V_p. First, we compute the self attention scores of all future representations to all past representations:

$$\text{unnormalized\_scores\_p} = \frac{Q_p K_o^T}{\sqrt{d_k}}. \tag{6}$$

Note, however, that the self-attention scores of each predicted future state with itself is not calculated– due to the key matrix being from the original states. Therefore, the superdiagonal needs to be replaced with the self-attention scores of each predicted future state with itself to achieve the attention score matrix shown in Equation 4. The details of this implementation are provided in Section D.4.

6 Experimentation

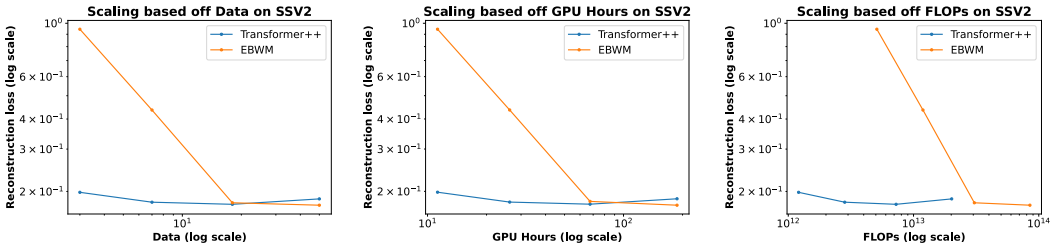
6.1 Computer Vision

For all experiments below, we utilized a frozen DINOv2 backbone [7] to encode all images into 768 dimensional features. Then, using a reconstruction objective (the same one discussed in Section 5.1), the models are trained to predict the feature of the next image conditioned on all past features.

Scaling on the Something-Something V2 (SSV2) dataset [96] for data, GPU hours, and FLOPs are shown in Fig. 3. We observe TAMs overfitting past the third scale, therefore, we also investigate the scaling for data, GPU hours, and FLOPs on an aggregated dataset of Kinetics-400 [97] and SSV2 in Fig. 4. EBWM consistently outperforms TAMs in data efficiency at higher scales, performs comparably or better in scaling with GPU hours, and performs worse in scaling with FLOPs. The experiments for NLP are in Section A.

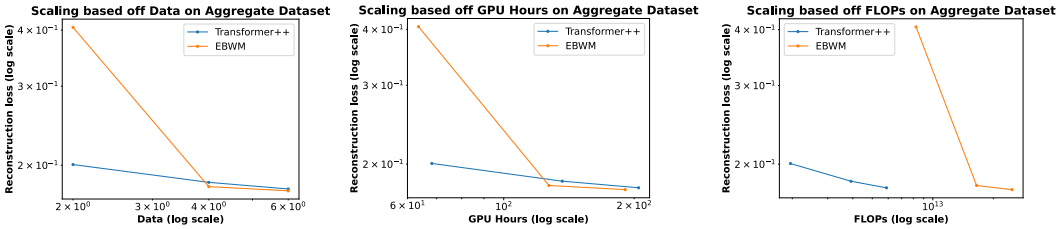
7 Discussion

Our experimental results demonstrate that initially EBWM scales slower than TAMs, but as scale increases, it matches and eventually exceeds the performance of TAMs in data and GPU hour efficiency. This outcome is promising for higher compute regimes, as the scaling rate of EBWM is higher than TAMs as computation increases. The slower initial scaling of EBWM is primarily due to the requirement of learning to produce representations for generating gradients to predict the next state, which is more involved than directly predicting the next state. Additionally, as there is high temporal consistency in CV, simply copying the most recent embedding, which TAMs can easily



(a) Scaling for data across epochs. (b) Scaling for A100 GPU Hours. (c) Scaling for FLOPs.

Figure 3: The reconstruction loss across different scales of data, GPU hours, and FLOPs (lower and farther to the left curves are better). The third datapoint for TAMs was the minimum loss value achieved during training–meaning from then onwards the model overfits the data. Note that the rate the loss decreased for EBWM is high initially due to being conditioned on random noise, whereas a simple copying of the most recent embedding for TAMs would yield a low reconstruction loss due to the nature of videos having high temporal consistency. Across runs, we consistently observe that EBWM is less susceptible to overfitting. The lowest validation loss was achieved by EBWM.



(a) Scaling for data across epochs. (b) Scaling for A100 GPU Hours. (c) Scaling for FLOPs.

Figure 4: The reconstruction loss across different scales of data, GPU hours, and FLOPs (lower and farther to the left curves are better) for the aggregated Kinetics-400 and SSV2 dataset. The results demonstrate the scalability of EBWM when compared to TAMs, achieving better scaling in both data and compute hours.

do, yields decent performance. Throughout our experiments we see that EBWM does not overfit the data, but TAMs do (Fig. 3). This reduced susceptibility to overfitting is due to EBMs learning a joint distribution, rather than a conditional distribution as in TAMs.

Although we do a side-by-side comparison of the scaling performance of EBWM with TAMs throughout this work, we do not see EBWM as a drop-in replacement for TAMs. Rather, having the four aspects of human cognition described, we see EBWM as different and even complementary to TAMs (Section B.5). There exist several current real-world use cases, such as low-latency LLM serving, where doing a single forward pass is sufficient, and where the added inference overhead of backpropagating the gradient when using EBWM would not be worth the extra computation. However, we also envision a world in which people needing long-term System 2 thinking to solve challenging problems use EBWM. How much compute would it be worth dedicating to prove a long-standing mathematical conjecture? Similarly, in cases where a wide search is done, for example when solving challenging coding problems as in AlphaCode 2 [88] and producing millions of potential solutions, using EBWM for the evaluation and improvement of generated states could vastly improve performance. EBWM offers a promising path towards neural architectures capable of achieving human-like cognition.

8 Conclusion

In this work, we proposed EBWM, a novel training approach for autoregressive world models involving predictions made in the input space rather than in the output space through the usage of an EBM. Structuring the architecture in this manner offers distinct benefits for achieving human-like cognitive capabilities. We demonstrate the scalability of EBWM through comprehensive performance

comparisons with TAMs in both Vision and NLP. Based on our promising experimental results, we believe EBWM provides a potential path towards achieving highly sought after human cognitive capabilities such as System 2 thinking or intelligent search. Our architecture has a few limitations that will be exciting future work to pursue:

- **Additional hyperparameters.** Due to the usage of MCMC for generation, EBWM involves additional hyperparameters such as the MCMC step size and the number of MCMC steps.
- **Further Scaling.** Due to our limited computational resources, we did not investigate the scaling of EBWM above $1,000$ A100 GPU hour runs, leaving the scaling trends for larger training runs unexplored. However, the experiments conducted show the rate of scaling for EBWM generally being higher than in TAMs as scale increases, offering promise towards further compute regimes.

References

[1] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023.

[2] Bo Peng, Eric Alcaide, Quentin Anthony, Alon Albalak, Samuel Arcadinho, Huanqi Cao, Xin Cheng, Michael Chung, Matteo Grella, Kranthi Kiran GV, et al. Rwkv: Reinventing rnns for the transformer era. *arXiv preprint arXiv:2305.13048*, 2023.

[3] Tobias Höppe, Arash Mehrjou, Stefan Bauer, Didrik Nielsen, and Andrea Dittadi. Diffusion models for video prediction and infilling, 2022.

[4] Yutao Sun, Li Dong, Shaohan Huang, Shuming Ma, Yuqing Xia, Jilong Xue, Jianyong Wang, and Furu Wei. Retentive network: A successor to transformer for large language models, 2023.

[5] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[6] Adrien Bardes, Jean Ponce, and Yann LeCun. Mc-jepa: A joint-embedding predictive architecture for self-supervised learning of motion and content features, 2023.

[7] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2023.

[8] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners, 2021.

[9] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers, 2021.

[10] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020.

[11] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations, 2020.

[12] OpenAI. Gpt-4 technical report, 2023.

[13] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jian-Yun Nie, and Ji-Rong Wen. A survey of large language models, 2023.

[14] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared 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 M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners, 2020.

[15] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob Menick, Sebastian Borgeaud, Andrew Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karen Simonyan. Flamingo: a visual language model for few-shot learning, 2022.

[16] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.

[17] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models, 2023.

[18] Roi Benita, Michael Elad, and Joseph Keshet. Diffar: Denoising diffusion autoregressive model for raw speech waveform generation. *arXiv preprint arXiv:2310.01381*, 2023.

[19] Po-chun Hsu, Da-Rong Liu, Andy T Liu, and Hung-yi Lee. Parallel synthesis for autoregressive speech generation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2023.

[20] Zalán Borsos, Raphaël Marinier, Damien Vincent, Eugene Kharitonov, Olivier Pietquin, Matt Sharifi, Dominik Roblek, Olivier Teboul, David Grangier, Marco Tagliasacchi, and Neil Zeghidour. Audiolm: a language modeling approach to audio generation, 2023.

[21] Shakked Noy and Whitney Zhang. Experimental evidence on the productivity effects of generative artificial intelligence. *Science*, 381(6654):187–192, 2023.

[22] Jared Spataro. Introducing microsoft 365 copilot – your copilot for work, March 2023. URL https://blogs.microsoft.com/blog/2023/03/16/introducing-microsoft-365-copilot-your-copilot-for-work/.

[23] URL https://stability.ai/news/stable-video-diffusion-open-ai-video-model.

[24] URL https://machinelearning.apple.com/research/hey-siri.

[25] OpenAI. Chatgpt can now see, hear, and speak, September 2023. URL https://openai.com/index/chatgpt-can-now-see-hear-and-speak/.

[26] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[27] Reza Mahjourian, Martin Wicke, and Anelia Angelova. Geometry-based next frame prediction from monocular video. In *2017 IEEE Intelligent Vehicles Symposium (IV)*, pages 1700–1707. IEEE, 2017.

[28] Yufan Zhou, Haiwei Dong, and Abdulmotaleb El Saddik. Deep learning in next-frame prediction: A benchmark review. *IEEE Access*, 8:69273–69283, 2020.

[29] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[30] Md Mofijul Islam, Alexi Gladstone, Riashat Islam, and Tariq Iqbal. Eqa-mx: Embodied question answering using multimodal expression. In *The Twelfth International Conference on Learning Representations*, 2023.

[31] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

[32] Yutong Bai, Xinyang Geng, Karttikeya Mangalam, Amir Bar, Alan Yuille, Trevor Darrell, Jitendra Malik, and Alexei A Efros. Sequential modeling enables scalable learning for large vision models. *arXiv preprint arXiv:2312.00785*, 2023.

[33] Anthony Hu, Lloyd Russell, Hudson Yeo, Zak Murez, George Fedoseev, Alex Kendall, Jamie Shotton, and Gianluca Corrado. Gaia-1: A generative world model for autonomous driving, 2023.

[34] Riashat Islam, Hongyu Zang, Manan Tomar, Aniket Didolkar, Md Mofijul Islam, Samin Yeasar Arnob, Tariq Iqbal, Xin Li, Anirudh Goyal, Nicolas Heess, et al. Representation learning in deep rl via discrete information bottleneck. *arXiv preprint arXiv:2212.13835*, 2022.

[35] Mohammad Samin Yasar and Tariq Iqbal. Vader: Vector-quantized generative adversarial network for motion prediction. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3827–3834. IEEE, 2023.

[36] Ann Lee, Peng-Jen Chen, Changhan Wang, Jiatao Gu, Sravya Popuri, Xutai Ma, Adam Polyak, Yossi Adi, Qing He, Yun Tang, et al. Direct speech-to-speech translation with discrete units. *arXiv preprint arXiv:2107.05604*, 2021.

[37] Rafael Valle, Kevin Shih, Ryan Prenger, and Bryan Catanzaro. Flowtron: an autoregressive flow-based generative network for text-to-speech synthesis. *arXiv preprint arXiv:2005.05957*, 2020.

[38] Curtis Hawthorne, Andrew Jaegle, Cătălina Cangea, Sebastian Borgeaud, Charlie Nash, Mateusz Malinowski, Sander Dieleman, Oriol Vinyals, Matthew Botvinick, Ian Simon, Hannah Sheahan, Neil Zeghidour, Jean-Baptiste Alayrac, João Carreira, and Jesse Engel. General-purpose, long-context autoregressive modeling with perceiver ar, 2022.

[39] OpenAI. Sora: first impressions, March 2024. URL https://openai.com/index/sora-first-impressions/.

[40] Karthik Valmeekam, Alberto Olmo, Sarath Sreedharan, and Subbarao Kambhampati. Large language models still can't plan (a benchmark for llms on planning and reasoning about change). *arXiv preprint arXiv:2206.10498*, 2022.

[41] Jinman Zhao and Xueyan Zhang. Exploring the limitations of large language models in compositional relation reasoning. *arXiv preprint arXiv:2403.02615*, 2024.

[42] Chengxuan Li, Di Huang, Zeyu Lu, Yang Xiao, Qingqi Pei, and Lei Bai. A survey on long video generation: Challenges, methods, and prospects. *arXiv preprint arXiv:2403.16407*, 2024.

[43] Bo Liu, Yuqian Jiang, Xiaohan Zhang, Qiang Liu, Shiqi Zhang, Joydeep Biswas, and Peter Stone. Llm+ p: Empowering large language models with optimal planning proficiency. *arXiv preprint arXiv:2304.11477*, 2023.

[44] Philipp Mondorf and Barbara Plank. Beyond accuracy: Evaluating the reasoning behavior of large language models–a survey. *arXiv preprint arXiv:2404.01869*, 2024.

[45] Kush Agrawal. To study the phenomenon of the moravec's paradox. *arXiv preprint arXiv:1012.3148*, 2010.

[46] Peter West, Ximing Lu, Nouha Dziri, Faeze Brahman, Linjie Li, Jena D Hwang, Liwei Jiang, Jillian Fisher, Abhilasha Ravichander, Khyathi Chandu, et al. The generative ai paradox:" what it can create, it may not understand". *arXiv preprint arXiv:2311.00059*, 2023.

[47] Patricia A Alexander. The development of expertise: The journey from acclimation to proficiency. *Educational researcher*, 32(8):10–14, 2003.

[48] Andreja Bubic, D Yves Von Cramon, and Ricarda I Schubotz. Prediction, cognition and the brain. *Frontiers in human neuroscience*, 4:1094, 2010.

[49] D. Tomasi, Gene-Jack Wang, Y. Studentsova, and N. Volkow. Dissecting neural responses to temporal prediction, attention, and memory: Effects of reward learning and interoception on time perception. *Cerebral cortex*, 25 10:3856–67, 2015. doi: 10.1093/cercor/bhu269.

[50] D. Schacter, D. Addis, and R. Buckner. Remembering the past to imagine the future: the prospective brain. *Nature Reviews Neuroscience*, 8:657–661, 2007. doi: 10.1038/nrn2213.

[51] Yanping Huang and Rajesh PN Rao. Predictive coding. *Wiley Interdisciplinary Reviews: Cognitive Science*, 2(5):580–593, 2011.

[52] Aran Nayebi, R. Rajalingham, M. Jazayeri, and G. R. Yang. Neural foundations of mental simulation: Future prediction of latent representations on dynamic scenes. *ArXiv*, 2023. doi: 10.48550/arXiv.2305.11772.

[53] L. Connell and Mark T. Keane. A model of plausibility. *Cognitive science*, 30 1:95–120, 2006. doi: 10.1207/s15516709cog0000_53.

[54] E. Brown and M. Brüne. The role of prediction in social neuroscience. *Frontiers in Human Neuroscience*, 6, 2012. doi: 10.3389/fnhum.2012.00147.

[55] Daniel Kahneman. *Thinking, fast and slow*. macmillan, 2011.

[56] Joaquin M Fuster. The prefrontal cortex and its relation to behavior. *Progress in brain research*, 87: 201–211, 1991.

[57] Nicolas P Rougier, David C Noelle, Todd S Braver, Jonathan D Cohen, and Randall C O'Reilly. Prefrontal cortex and flexible cognitive control: Rules without symbols. *Proceedings of the National Academy of Sciences*, 102(20):7338–7343, 2005.

[58] Christian Tomani, Kamalika Chaudhuri, Ivan Evtimov, Daniel Cremers, and Mark Ibrahim. Uncertainty-based abstention in llms improves safety and reduces hallucinations. *arXiv preprint arXiv:2404.10960*, 2024.

[59] A. Peters, B. McEwen, and Karl J. Friston. Uncertainty and stress: Why it causes diseases and how it is mastered by the brain. *Progress in Neurobiology*, 156:164–188, 2017. doi: 10.1016/j.pneurobio.2017.05.004.

[60] I. Vilares, J. D. Howard, Hugo L. Fernandes, J. Gottfried, and Konrad Paul Kording. Differential representations of prior and likelihood uncertainty in the human brain. *Current Biology*, 22:1641–1648, 2012. doi: 10.1016/j.cub.2012.07.010.

[61] Issidoros C. Sarinopoulos, D. Grupe, Kristen L. Mackiewicz, J. Herrington, M. Lor, E. E. Steege, and J. Nitschke. Uncertainty during anticipation modulates neural responses to aversion in human insula and amygdala. *Cerebral cortex*, 20 4:929–40, 2010. doi: 10.1093/cercor/bhp155.

[62] 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, 2017.

[63] Yann LeCun. A path towards autonomous machine intelligence version 0.9. 2, 2022-06-27. *Open Review*, 62, 2022.

[64] Yann LeCun, 2024. URL https://www.linkedin.com/posts/yann-lecun_lots-of-confusion-about-what-a-world-model-activity-7165738293223931904-vdgR.

[65] David Ha and Jürgen Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2018.

[66] Hao Liu, Wilson Yan, Matei Zaharia, and Pieter Abbeel. World model on million-length video and language with ringattention. *arXiv preprint arXiv:2402.08268*, 2024.

[67] Manfred Morari and Jay H Lee. Model predictive control: past, present and future. *Computers & chemical engineering*, 23(4-5):667–682, 1999.

[68] Tadahiro Taniguchi, Shingo Murata, Masahiro Suzuki, Dimitri Ognibene, Pablo Lanillos, Emre Ugur, Lorenzo Jamone, Tomoaki Nakamura, Alejandra Ciria, Bruno Lara, et al. World models and predictive coding for cognitive and developmental robotics: frontiers and challenges. *Advanced Robotics*, 37(13): 780–806, 2023.

[69] Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023.

[70] Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

[71] Adrien Bardes, Quentin Garrido, Jean Ponce, Xinlei Chen, Michael Rabbat, Yann LeCun, Mido Assran, and Nicolas Ballas. V-jepa: Latent video prediction for visual representation learning. 2023.

[72] Tobias Höppe, Arash Mehrjou, Stefan Bauer, Didrik Nielsen, and Andrea Dittadi. Diffusion models for video prediction and infilling. *arXiv preprint arXiv:2206.07696*, 2022.

[73] Dirk Weissenborn, Oscar Täckström, and Jakob Uszkoreit. Scaling autoregressive video models. *arXiv preprint arXiv:1906.02634*, 2019.

[74] Jiasen Lu, Christopher Clark, Sangho Lee, Zichen Zhang, Savya Khosla, Ryan Marten, Derek Hoiem, and Aniruddha Kembhavi. Unified-io 2: Scaling autoregressive multimodal models with vision, language, audio, and action, 2023.

[75] Chang Chen, Yi-Fu Wu, Jaesik Yoon, and Sungjin Ahn. Transdreamer: Reinforcement learning with transformer world models. *arXiv preprint arXiv:2202.09481*, 2022.

[76] Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. *arXiv preprint arXiv:1912.01603*, 2019.

[77] Mengjiao Yang, Yilun Du, Kamyar Ghasemipour, Jonathan Tompson, Dale Schuurmans, and Pieter Abbeel. Learning interactive real-world simulators, 2023.

[78] Siddique Latif, Aun Zaidi, Heriberto Cuayahuitl, Fahad Shamshad, Moazzam Shoukat, and Junaid Qadir. Transformers in speech processing: A survey. *arXiv preprint arXiv:2303.11607*, 2023.

[79] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

[80] Hao Zou, Zae Myung Kim, and Dongyeop Kang. A survey of diffusion models in natural language processing. *arXiv preprint arXiv:2305.14671*, 2023.

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

[82] Sachin Goyal, Ziwei Ji, Ankit Singh Rawat, Aditya Krishna Menon, Sanjiv Kumar, and Vaishnavh Nagarajan. Think before you speak: Training language models with pause tokens. *arXiv preprint arXiv:2310.02226*, 2023.

[83] Benjamin Hoover, Yuchen Liang, Bao Pham, Rameswar Panda, Hendrik Strobelt, Duen Horng Chau, Mohammed Zaki, and Dmitry Krotov. Energy transformer. *Advances in Neural Information Processing Systems*, 36, 2024.

[84] Yezhen Wang, Tong Che, Bo Li, Kaitao Song, Hengzhi Pei, Yoshua Bengio, and Dongsheng Li. Your autoregressive generative model can be better if you treat it as an energy-based one. *arXiv preprint arXiv:2206.12840*, 2022.

[85] Sumanta Bhattacharyya, Amirmohammad Rooshenas, Subhajit Naskar, Simeng Sun, Mohit Iyyer, and Andrew McCallum. Energy-based reranking: Improving neural machine translation using energy-based models. *arXiv preprint arXiv:2009.13267*, 2020.

[86] Anton Bakhtin, Yuntian Deng, Sam Gross, Myle Ott, Marc'Aurelio Ranzato, and Arthur Szlam. Residual energy-based models for text. *Journal of Machine Learning Research*, 22(40):1–41, 2021.

[87] Gala Stojnić, Kanishk Gandhi, Shannon Yasuda, Brenden M Lake, and Moira R Dillon. Commonsense psychology in human infants and machines. *Cognition*, 235:105406, 2023.

[88] AlphaCode Team. Alphacode 2 technical report. December 2023.

[89] Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models, 2023.

[90] Meta. Introducing meta llama 3: The most capable openly available llm to date, April 2024. URL https://ai.meta.com/blog/meta-llama-3/.

[91] Jonathan St BT Evans. In two minds: dual-process accounts of reasoning. *Trends in cognitive sciences*, 7 (10):454–459, 2003.

[92] Adam L Alter, Daniel M Oppenheimer, Nicholas Epley, and Rebecca N Eyre. Overcoming intuition: metacognitive difficulty activates analytic reasoning. *Journal of experimental psychology: General*, 136 (4):569, 2007.

[93] Geoffrey E Hinton. Training products of experts by minimizing contrastive divergence. *Neural computation*, 14(8):1771–1800, 2002.

[94] Yilun Du, Shuang Li, Joshua Tenenbaum, and Igor Mordatch. Improved contrastive divergence training of energy based models. *arXiv preprint arXiv:2012.01316*, 2020.

[95] Ze Wang, Jiang Wang, Zicheng Liu, and Qiang Qiu. Energy-inspired self-supervised pretraining for vision models. *arXiv preprint arXiv:2302.01384*, 2023.

[96] Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, et al. The" something something" video database for learning and evaluating visual common sense. In *Proceedings of the IEEE international conference on computer vision*, pages 5842–5850, 2017.

[97] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017.

[98] Together Computer. Redpajama: an open dataset for training large language models, 2023. URL https://github.com/togethercomputer/RedPajama-Data.

[99] Yongshuo Zong, Oisin Mac Aodha, and Timothy Hospedales. Self-supervised multimodal learning: A survey. *arXiv preprint arXiv:2304.01008*, 2023.

[100] Mohammad Samin Yasar, Md Mofijul Islam, and Tariq Iqbal. Imprint: Interactional dynamics-aware motion prediction in teams using multimodal context. *ACM Transactions on Human-Robot Interaction*, 2022.

[101] Md Mofijul Islam, Mohammad Samin Yasar, and Tariq Iqbal. Maven: A memory augmented recurrent approach for multimodal fusion. *IEEE Transactions on Multimedia*, 2022.

[102] Chameleon Team. Chameleon: Mixed-modal early-fusion foundation models, 2024.

[103] Md Mofijul Islam and Tariq Iqbal. Mumu: Cooperative multitask learning-based guided multimodal fusion. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pages 1043–1051, 2022.

[104] Md Mofijul Islam and Tariq Iqbal. Hamlet: A hierarchical multimodal attention-based human activity recognition algorithm. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10285–10292. IEEE, 2020.

[105] Md Mofijul Islam and Tariq Iqbal. Multi-gat: A graphical attention-based hierarchical multimodal representation learning approach for human activity recognition. *IEEE Robotics and Automation Letters*, 6(2):1729–1736, 2021.

[106] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36, 2024.

[107] Lukas Berglund, Meg Tong, Max Kaufmann, Mikita Balesni, Asa Cooper Stickland, Tomasz Korbak, and Owain Evans. The reversal curse: Llms trained on" a is b" fail to learn" b is a". *arXiv preprint arXiv:2309.12288*, 2023.

[108] Dwarkesh Patel. Will scaling work?, December 2023. URL https://www.dwarkeshpatel.com/p/will-scaling-work.

[109] Pablo Villalobos, Jaime Sevilla, Lennart Heim, Tamay Besiroglu, Marius Hobbhahn, and Anson Ho. Will we run out of data? an analysis of the limits of scaling datasets in machine learning. *arXiv preprint arXiv:2211.04325*, 2022.

[110] Suriya Gunasekar, Yi Zhang, Jyoti Aneja, Caio César Teodoro Mendes, Allie Del Giorno, Sivakanth Gopi, Mojan Javaheripi, Piero Kauffmann, Gustavo de Rosa, Olli Saarikivi, et al. Textbooks are all you need. *arXiv preprint arXiv:2306.11644*, 2023.

[111] Meta. Our responsible approach to meta ai and meta llama 3, April 2024. URL https://ai.meta.com/blog/meta-llama-3-meta-ai-responsibility/.

[112] Md Mofijul Islam, Reza Mirzaiee, Alexi Gladstone, Haley Green, and Tariq Iqbal. Caesar: An embodied simulator for generating multimodal referring expression datasets. *Advances in Neural Information Processing Systems*, 35:21001–21015, 2022.

[113] Md Mofijul Islam, Alexi Gladstone, and Tariq Iqbal. Patron: perspective-aware multitask model for referring expression grounding using embodied multimodal cues. In *Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence and Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence and Thirteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'23/IAAI'23/EAAI'23. AAAI Press, 2023. ISBN 978-1-57735-880-0. doi: 10.1609/aaai.v37i1.25177. URL https://doi.org/10.1609/aaai.v37i1.25177.

[114] Trieu H Trinh, Yuhuai Wu, Quoc V Le, He He, and Thang Luong. Solving olympiad geometry without human demonstrations. *Nature*, 625(7995):476–482, 2024.

[115] Nicholas Lee, Thanakul Wattanawong, Sehoon Kim, Karttikeya Mangalam, Sheng Shen, Gopala Anumanchipali, Michael W Mahoney, Kurt Keutzer, and Amir Gholami. Llm2llm: Boosting llms with novel iterative data enhancement. *arXiv preprint arXiv:2403.15042*, 2024.

[116] Yann LeCun, Sumit Chopra, Raia Hadsell, M Ranzato, and Fujie Huang. A tutorial on energy-based learning. *Predicting structured data*, 1(0), 2006.

[117] Yang Song and Diederik P Kingma. How to train your energy-based models. *arXiv preprint arXiv:2101.03288*, 2021.

[118] Yilun Du, Shuang Li, Yash Sharma, Josh Tenenbaum, and Igor Mordatch. Unsupervised learning of compositional energy concepts. *Advances in Neural Information Processing Systems*, 34:15608–15620, 2021.

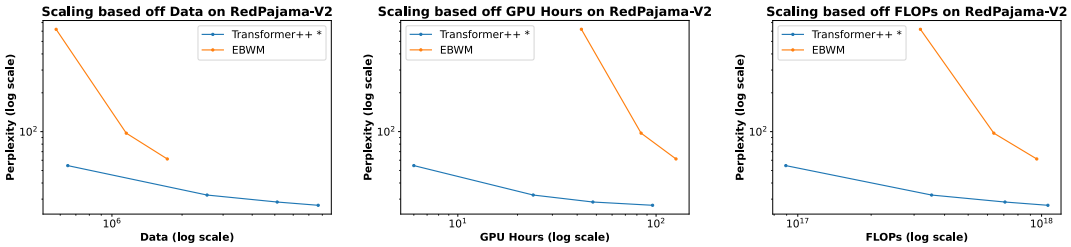
[119] Yilun Du, Shuang Li, and Igor Mordatch. Compositional visual generation with energy based models. *Advances in Neural Information Processing Systems*, 33:6637–6647, 2020.

[120] Sid Black, Stella Biderman, Eric Hallahan, Quentin Anthony, Leo Gao, Laurence Golding, Horace He, Connor Leahy, Kyle McDonell, Jason Phang, Michael Pieler, USVSN Sai Prashanth, Shivanshu Purohit, Laria Reynolds, Jonathan Tow, Ben Wang, and Samuel Weinbach. Gpt-neox-20b: An open-source autoregressive language model, 2022.

[121] William A Falcon. Pytorch lightning. *GitHub*, 3, 2019.

A Additional Experimentation: EBWM for Natural Language Processing

For all experiments in Natural Language Processing (NLP), we utilize the RedPajama-V2 dataset [98] $100B$ sample from HuggingFace, with a manually created train and validation split of $65,818,073$ and $330,745$ respectively. The training objective is the traditional language modeling objective of predicting the next token in a sequence. We report scaling for data, GPU hours, and FLOPs of EBWM compared to TAMs in Fig. 5. Despite not being able to tune hyperparameters due to limited computational resources, the results demonstrate the scalability of EBWM in data efficiency.



(a) Scaling for data across epochs. (b) Scaling for A100 GPU Hours. (c) Scaling for FLOPs.

Figure 5: The perplexity across different scales of data, GPU hours, and FLOPs (lower and farther to the left curves are better) for the RedPajama-V2 dataset. The experimental results show the promising scaling of NLP models using EBWM. We believe that with more resources to search over different hyperparameters EBWM could perform significantly better. EBWM used a smaller batch size due to involving more memory, and hence was much slower, resulting in it having less datapoints given our compute limit. *The first two points for the Transformer++ were perplexity values on the training set.

B Future Works and Broader Impact

EBWM, being qualitatively different from existing autoregressive architectures, while having all four cognitive facets described, opens several future research directions.

B.1 EBWM for other domains

We designed EBT as a modality agnostic autoregressive architecture. Consequently, we find it highly plausible EBWM could be used in domains other than NLP and CV. For example, sequential data in Audio Processing, Graph based learning, Time Series Analysis, and Tactile & Haptic Feedback, could be modeled similarly to CV/NLP data with EBWM.

B.2 Multimodal EBWM

Multimodal learning has progressed rapidly within recent years [99–106]. EBWM, being qualitatively different from existing architectures, offers an exciting future research direction for extensions to multiple modalities.

B.3 Reversal curse

Recently, a phenomenon known as "The Reversal Curse" has been observed where LLMs fail to learn a symmetric mapping [107] "B is A" despite learning "A is B". For example, LLMs trained on an example such as "Q: Who is Tom Cruise's mother? A: Mary Lee Pfeiffer" often fail to generalize to know the answer to the reverse question of "Who is Mary Lee Pfeiffer's son?" Remarkably, the Reversal Curse has manifested itself in LLMs regardless of the size or scale [107]—probing researchers to investigate whether there are fundamental limitations to TAMs. One predicted cause of The Reversal Curse is the nature of gradient updates being passed only to tokens within context. That is, while learning the mapping "A is B", none of B's tokens are within context meaning they do not receive gradient updates. In this paper, we hypothesize using EBWM would help reduce this phenomenon, as both A's and B's tokens are within context during gradient updates due to future

state predictions being made in the input space. Therefore, an exciting research direction would be investigating whether this hypothesis is correct in LLMs trained with EBWM.

B.4 World models for acting

In this work, we focus on a specific instance of world models where a default "no-op" action is taken. However, more complex world models often consist of forward rollouts conditioned off of specific actions being taken. In this paradigm, EBWM offers high promise due to the nature of EBMs. Particularly, given a model trained to estimate the unnormalized joint distribution of the current context, future, as well as the next action, such world models could implicitly be used as policies to generate actions to achieve a specific state. This would involve holding the current context constant and minimizing the energy by passing the gradient back to the action inputs and future state predictions. Thus, world models trained in this manner become capable of more than just predicting the future, but also in decision making to achieve a specific goal state.

B.5 Improving traditional autoregressive model predictions

As demonstrated in [86, 85], EBMs can be used to improve the quality of generated text from language models. In the context of EBWM, this is possible for world models across domains. This process would involve making a forward pass with a traditional autoregressive world model, passing this model's output into EBWM, and then using MCMC to further improve upon the TAMs prediction. This combination allows for fast inference when necessary, such as when low latency is a priority. It also allows for the cognitive facets described in Section 1, potentially enabling for a situation where different more challenging problems get the extra computational resources needed to solve them. This is one of the reasons we consider EBWM *complementary* to TAMs, rather than being a competitor.

B.6 Data Augmentation

One problem towards further scaling of foundation models is the scale of data [108, 109], as, especially in the realm of NLP, much of the data on the open internet has been exhausted. As a consequence the creation of synthetic datasets has become common [110–113]. With TAMs, the process of data augmentation usually involves taking advantage of algorithmically verifiable text such as code or math [114] or synonym replacement [115] to generate plausible next tokens. With EBWM, the process of data augmentation is significantly easier due to the approximation of a joint distribution $p(x_1, ..., x_t, x_{t+1})$ rather than the conditional distribution $p(x_{t+1}|x_1, ..., x_t)$ TAMs approximate. This was demonstrated in Section 5.2, where a different number of MCMC steps were used, as each of these MCMC steps can be seen as a different data sample ($p(x_1, ..., x_t, \hat{x}_{t+1)}$) due to there being a different condition ($\hat{x}_{t+1)}$). A promising future direction is an investigation of the extent to which data augmentation with EBWM can be done without reducing performance, and the benefits this can have in reducing the amount of pre-augmentation data needed.

B.7 Intelligent Search

Due to a lack of computational resources, we were unable to train models in the regime of $1,000$ or more A100 GPU Hours. Therefore, the usage of EBWM for intelligent search remains untested at scale. We leave it to future work to further scale with more GPUs and investigate the qualitative differences in searching the state space with TAMs versus with EBWM as described in Section 4.

B.8 Societal impact

The ability to achieve human level cognition with AI offers benefits in multiple domains. As such, EBWM offers several potential positive impacts, through the enabling of AI to be more similar to human cognition. On the other hand, it's also possible that more intelligent AI models trained using EBWM could be misused for harm by malicious actors.

C Ablation Studies

C.1 Losses

In addition to EBM training over sequences using a reconstruction loss [95], we develop a new approach involving the prediction of ground truth energy labels–we call this Regressive Energy Based Modeling. This allows us to model energy-based model training as a regression task:

$$\mathcal{L}_e = D(E, \hat{E}) \tag{7}$$

Where $\mathcal{L}_e$ denotes the energy loss. To generate these ground truth energy labels, we use the distance between the embeddings/features of a ground truth and predicted future state.

$$E = D(\mathbf{z}, \hat{\mathbf{z}}). \tag{8}$$

As such, we call this Representation-Induced Labeling.

One benefit of using encoder representations for labeling the ground truth energy value of a given future prediction is the ability to regularize the output energy space. Particularly, since we use the cosine similarity as a distance function, labeled energy values by default are within the range $[-1, 1]$. In this work, we further scale this range to be $[0, 1]$, with lower energy values indicating higher compatibility future state predictions.

Utilizing this characteristic, we add an additional regularization term, which we call the Out-of-Bounds loss. This regularization term, in addition to the energy loss term, enforces predicted energies to be within a specified range, in this case $[0, 1]$:

$$\mathcal{L}_b = \max(0, \hat{E} - 1) + \max(0, -\hat{E})$$

The results for using these two losses are shown in Table 2. Overall, we find these losses are **not** helpful in increasing stability, and that the rather simple approach of a single reconstruction objective works best.

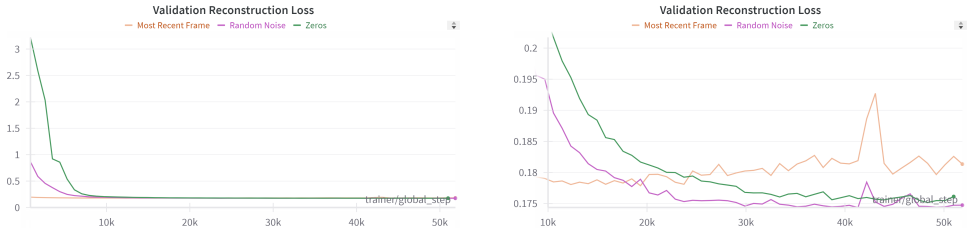
C.2 Condition for MCMC

Energy Based Models are a family of generative models that predict the unnormalized density, or compatibility, of a configuration of variables. Through the training a robust energy space, inference from Energy Based Models involves traversal of this energy space through gradient descent. Consequentially, one of the most common sampling techniques for Energy Based Models is Markov Chain Monte Carlo (MCMC). At each time step of the Markov Chain, the current predicted $\hat{y}$ is inputted into the model, an energy is computed, and through gradient descent this energy function is descended. Historically MCMC has begun from random noise [116, 117], resulting in a very long Markov Chain to reach convergence. However, recently works have experimented with MCMC from a condition [118, 119, 95]. This has the advantage of reducing the number of steps to convergence within the energy function. We experimented with a couple of different approaches for our MCMC condition in CV including conditioning on the most recent recent frame, conditioning on all zeros, and conditioning from random noise. We found that although conditioning on the most recent frame achieves better initial convergence due to the high temporal consistency in videos, random noise scaled better. Plots for losses based on different conditions are shown in Fig. 6.

D Additional EBWM Details

Tables 3 and 4 specify model information and hyperparameters for each experiment.

The effective batch size for each training run is triple the batch size per GPU, as we train with three Nvidia A100 GPUs for all experiments. Note that although we specify for the models to train for 400 epochs, this was not completed due to having a limited compute budget. We use zero dropout across



(a) Earlier validation scaling.

(b) Later validation scaling.

Figure 6: Validation loss curves with different MCMC conditions. Although the loss is initially better when conditioning on the most recent frame due to a high temporal overlap in videos, conditioning on random noise ultimately achieves the best validation loss. The validation loss of conditioning on the most recent frame starts at 0.196.

Table 3: **Hyperparameters for TAMs.** Model information and hyperparameters for our TAMs experiments.

Hyperparameter	**CV**	**NLP**
Batch Size per GPU	64	128
Effective Batch Size	192	384
Epochs	400	
Optimizer	AdamW	
Optimizer Momentum	$\beta_1, \beta_2 = 0.9, 0.999$	
Freeze Encoder Epochs	2000	
Learning Rate (LR)	$1e-3$	$6e-4$
LR Schedule	Linear warm up cosine decay	
Warmup steps	$1e4$	
Warmup Base LR Divider	20	
Minimum LR Scale	10	
Gradient Clip Value	1	
Transformer Blocks	12	
Multi-headed Attention Heads	12	
Weight Decay	0.01	
Context Length	16	256
ViT Backbone Model	DINOv2 [7]	-
ViT Backbone Size	Base	-
Image Dimension	224x224	-
Tokenizer	-	EleutherAI/gpt-neox-20b [120]
Vocab Size	-	50277

all training, and regularize model weights by using weight decay. To schedule the learning rate, we use a linear warm up with cosine annealing. We warm up for $1e4$ steps and decay by at most $10x$ the base learning rate. We set the coefficients for energy loss and out-of-bounds loss to zero, due to the results found in Table 2, removing them from the loss calculation.

We utilized the Llama 2 transformer implementation [17] for TAMs and used this implementation as the backbone upon which we built EBT. We use the following rule for scaling all learning rates (all learning rates shown in the table were base learning rates that got scaled according to this rule):

$$\text{lr} = \text{base\_learning\_rate} * \text{effective\_batch\_size}/256$$

Where the effective_batch_size is calculated based off the batch size per GPU and the number of GPUs multiplied (since we use DDP).

Table 4: **Hyperparameters for EBWM experiments.** Model information and hyperparameters for our EBWM experiments.

Hyperparameter	CV	NLP
Batch Size per GPU	64	24
Effective Batch Size	192	72
Epochs	400	
Optimizer	AdamW	
Optimizer Momentum	$\beta_1, \beta_2 = 0.9, 0.999$	
Learning Rate (LR)	$2e-4$	
LR Schedule	Linear warm up cosine decay	
Warmup Steps	$1e4$	
Warmup Base LR Divider	20	
Minimum LR Scale	10	
Energy Loss Coefficient	0	
Out-of-bounds Loss Coefficient	0	
Reconstruction Coefficient	60	
MCMC Steps	4	2
MCMC Step Size	$3e4$	$3e5$
Learnable MCMC Step Size	✓	
MCMC Step Size LR Multiplier	$2e5$	$2e6$
Clamp Future Gradient	✓	
Langevin Dynamics Noise	0	
Weight Decay	0.01	
Context Length	16	256
Transformer Blocks	12	
Multi-headed Attention Heads	12	
Gradient Clip Value	1	
End MLP Layers	1	
ViT Backbone Model	DINOv2 [7]	-
ViT Backbone Size	Base	-
Image Dimension	224x224	-
Tokenizer	-	EleutherAI/gpt-neox-20b [120]
Vocab Size	-	50277

D.1 Reproducibility

We utilize a singularity container and PyTorch Lightning [121] for all experiments. We seed all libraries using PyTorch Lightning with a seed of 33. We plan to release all code, containers for execution, as well as a comprehensive setup guide in the future. All scaling experiments conducted within the main paper were done with three Nvidia A100 80 GB GPU's for at most 72 hours (for an effective maximum amount of 216 GPU hours). We plan to release all source code on GitHub publicly soon.

D.2 Improving Stability

We experiment with several different hyperparameters to increase the stability of the EBWM approach.

D.2.1 Clamping Gradients

We clamp the future gradient for MCMC to help reconstruct images and avoid exploding gradient on backward passes. As clamping gradients for regular backpropagation is common to prevent loss spikes, this was an intuitive and helpful inductive bias for improving stability. The results in Table 2 demonstrate that this choice improves stability.

D.2.2 High MCMC step size

In addition to setting a high MCMC step size, we make the MCMC step size a learnable parameter. We calculate its learning rate by multiplying the model's learning rate by the MCMC step size learning rate multiplier. We find that the initial values for the MCMC step size have a large effect on the magnitude of gradients generated, and hence the stability of the model. Particularly, a smaller MCMC step size resulted in larger generated gradients. The results for a lower MCMC step size in Table 2 demonstrate this effect, with a smaller MCMC step size being less stable.

D.3 More intuition

Future state prediction in the input versus output space: We posit that there exists a fundamental distinction between the internal representations associated with model inputs and those concerning model outputs. Specifically, models generate internal representations *of* inputs, as these serve as the foundational elements that exclusively dictate the model's behavior at any given point in time. Conversely, since outputs do not affect a model's behavior at a given point in time, we contend that models construct representations *for predicting* such outputs. This distinction leads us to a consequential insight: models may not achieve a genuine understanding of outputs in the same way they understand inputs. Instead, they primarily excel in making predictions based on these output-focused representations. This foundational difference implies that while models may develop an intricate understanding of context when presented as input data, such understanding does not naturally extend to the realm of predictions made in the output space—indicating a limitation in models' abilities to comprehend future state predictions in the output space. This intuition further supports the principles behind EBWM.

D.4 Energy-Based Transformer Full Implementation

As described in Section 5.3, EBT involves two separate tensors–one for past states and one for predicted future states. We denote these as z_1^n and $\hat{z}_1^n$ where z are known past states and $\hat{z}$ are predicted future states. The intended attention scores matrix is the following:

$$\text{scores} = \begin{bmatrix} \alpha_{z_1,z_1} & \alpha_{z_1,\hat{z}_2} & 0 & \dots & 0 \\ \alpha_{z_2,z_1} & \alpha_{z_2,z_2} & \alpha_{z_2,\hat{z}_3} & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ \alpha_{z_n,z_1} & \alpha_{z_n,z_2} & \alpha_{z_3,z_3} & \dots & \alpha_{z_n,\hat{z}_{n+1}} \end{bmatrix}.$$

To compute Equation 4, we first need to append a column to the right side of the unnormalized_scores_p matrix, as the size of the matrix is currently $n-1 \times n-1$, but we need to have n representations within context. After doing this, we first mask out the superdiagonal, to ensure that the probabilities in the score matrix only correspond to the values of the predicted future states with itself. This masking operation is done through elementwise multiplication of a matrix with 1's everywhere except the superdiagonal, which has 0's. Then, we compute the self-attention scores of each predicted future state with itself, using the following equation:

$$\text{z\_p\_self\_attention} = sum(Q_p * K_p), \tag{9}$$

where the $*$ indicates the Hadamard product and the sum is across the fourth, attention head, dimension. Using a superdiagonal mask again, we set the diagonal of the unnormalized_scores_p to these values. Now, after applying the softmax:

$$\text{scores\_p} = \text{softmax}\left(\text{unnormalized\_scores\_p}\right), \tag{10}$$

we have the intended scores matrix shown in Equation 4. However, one more barrier towards finally extracting all updated z_{p1}^n representations is the fact that we cannot simply multiply this resulting scores matrix by the values matrix, as each element of the superdiagonal corresponds to a different predicted next future state. Thus, using similar techniques to before, we first clone and then extract the superdiagonal from this scores matrix using a diagonal mask.

After extracting the superdiagonal, we can multiply the resulting scores matrix by the V_o matrix to get all of the representations summed together of each predicted future state with all past states. This is represented as the following matrix multiplication:

$$z_{p1}^n = \text{scores\_p} \cdot V_o. \tag{11}$$

As we also need to add the representation of each predicted future state weighted with its own attention score (what was extracted on the superdiagonal), we perform another Hadamard product of the V_p matrix with the cloned superdiagonal to get these values, and then add these element wise to the z_{p1}^n representations. Now, we have computed the intended representations involving the scores matrix shown in Equation 4. Thus, z_1^n and $\hat{z}_1^n$ are updated using z_{o1}^n and z_{p1}^n respectively, by multiplying these tensors by the output weight matrix W_o .

E Counterarguments

E.1 System 2 thinking

In this paper, one of the major claims was that TAMs cannot do System 2 thinking like humans involving leveraging a dynamic amount of computation for predictions. However, there are common counterarguments to this, which we address, in hopes of clarifying why we believe this is not currently possible.

E.1.1 Chain-of-thought reasoning

First, chain-of-thought involves reasoning over a discrete state space, which limits the granularity of "thoughts". Second, chain-of-thought is not a capability internal to a model architecture, but rather something done externally over tokens. This capability should be built into an architecture and trained. Third, each individual token has a finite amount of computation to produce it, meaning that models cannot reason over each individual token used. Ideally, just as how when humans think "step by step," each step takes varying amounts of times, models could focus on each token for a specific amount of time until deemed adequate (as done in EBWM).

E.1.2 Pause token

Recently, seeking to achieve human-like reasoning capabilities researchers investigated the usage of a "pause token" [82]. Training models in this manner achieves better approximation towards cognitive facet (3) dynamic computation allocation 1. However, this approach is ultimately limited for the same reason that diffusion models are, in that they have no way of determining when a sufficient amount of computation has been used. This requires the ability to discriminate the plausibility of predictions, as described in Section 1.