DEBRA: A DISCRETE METHOD FOR ENERGY-BASED REASONING

Anonymous authors
Paper under double-blind review

ABSTRACT

Eenergy-based model (EBM) provides a unified framework to model data dependencies in various tasks (Lecun et al., 2006), among which the track of utilizing EBM in reasoning and planning tasks has been actively explored (Du et al. (2021; 2022); Chen et al. (2023); Du et al. (2024); Luo et al. (2024)). However, most of the existing energy-based and diffusion-style reasoning methods assume training and inference in a continuous space, which can potentially lead to suboptimal parameterization and inefficient training and sampling (Shi et al., 2025), as most of the contemporary reasoning and planning data are discrete, such as textual thoughts and robot motions ((Khatib, 1987; Wei et al., 2023)). To close the gap, we propose *Discrete Energy-Based ReAsoning*(Debra), a novel training and inference paradigm for solving discrete reasoning tasks such as Countdown and Sudoku. Inspired by the IRED method (Du et al., 2024) and the masked diffusion method Austin et al. (2023), Our method features with an annealed sequence of EBMs corresponding to an iterative unmasking schedule. Debra is novelly trained on a contrastively amplified discrete objective with pseudo-likelihood, and performs Gibbs sampling for inference, allowing for test-time scaling. Experiments on various discrete reasoning tasks show that our method achieves better or comparable performances compared to our autoregressive, diffusional, and energy-based counterparts, and is superior in smaller parameter sizes, faster converging rates, better generalizability, and more flexible choices of base models. To our knowledge, Debra is the first work to explicitly integrate EBM with fully discrete training for doing reasoning tasks, offering both theoretical insights and practical benefits for mainstream reasoning and planning applications. We release our code and checkpoints on GitHub.

1 INTRODUCTION

Treating reasoning and planning as discrete instead of continuous has its pragmatic necessity for any systems that need to *commit, act* and *communicate* under finite precision and time ((Turing, 1937; McCarthy & Hayes, 1969; Chemero, 2009)). Classical decision-making models, both biological and digital, expect an integration of continuous sensory evidence before crossing a threshold and forming a discrete decision ((Vogelsang et al., 2023)), and recent advancements of machine learning methods performing reasoning and planning in the categorical text and state spaces further support such statement ((Wei et al., 2023; Yang et al., 2023; Yao et al., 2023; Zhuang et al., 2024),).

Among them, the reasoning and planning methods utilizing Energy-Based Model (EBM) (Du et al. (2021; 2022); Chen et al. (2023); Du et al. (2024); Luo et al. (2024)) demonstrate a unique generality by formulating the various reasoning and planning tasks as an optimization problem over input $\boldsymbol{x}$ and output $\boldsymbol{y}$, with less parameterization bias and superior adaptivity to the out-of-domain data. However, most of the existing EBM methods assume a continuous energy space and adapt gradient-based training objectives and sampling methods to solve both continuous and discrete reasoning tasks. When dealing with discrete reasoning tasks, a soft prediction is first proposed, and then rounded for the final prediction ((Wang et al., 2022)). Moreover, these gradient-based methods requires multiple steps of gradient decent to reach an energy minima, meaning that the initial suboptimal solution is far from feasible ((Du et al., 2022; 2024)). A natural question raised is whether we can model the energy landscapes from the discrete data space more directly and efficiently.

In this work, we introduce a novel *Discrete Method for Energy-Based Reasoning* (Debra) which is analogous to the previous energy-based IRED method ((Du et al., 2024)) and the non-energy-based masked diffusion methods ((Austin et al., 2023; Shi et al., 2025)) in terms of using noise (masking) schedules during training and inference, but with cross-entropy loss as the major training objective. We believe that when combined with a EBM framework, the gradient-agnostic cross-entropy objective is both simpler to implement and more compatible to the current mainstream workhorses for discrete reasoning tasks, such as autoregressive models for language modeling and heuristic-based planners for decision making, rather than the delicate evidence lower bound mostly used in diffusion or autoencoder objectives ((Du et al., 2022; 2024; Shi et al., 2025).

Specifically, we formulate the discrete reasoning task as minimizing the energy function $E_\theta(\boldsymbol{x}, \boldsymbol{y})$, which takes in an input $\boldsymbol{x}$ and an output $\boldsymbol{y}$. Following the IRED work (Du et al., 2024), we break down the ultimate energy landscape we want to learn into a sequence of annealed EBMs which iteratively refines and guides our predicted $\boldsymbol{y}$ to the global minima. We define the sequential EBMs over the unmasking schedule, so that each EBM models the annealed energy landscape of the input and a partially unmasked output. To stabilize the joint training of the sequential EBMs and to augment the performance, we additionally use Hinge loss as the contrastive objective between the energy distributions of pairs of positive and negative samples. For inference, we adapt Gibbs sampling with our trained sequential EBMs to allow adjustable test-time computations for a refined inference performance.

We test the performance of Debra on various discrete reasoning tasks, including Countdown, Sudoku, Connectivity and Binary Satisfiability (Du et al., 2024; Ye et al., 2025), and compare with various energy-based and non-energy-based baselines under similar settings in parameterization, training and inference. Debra achieves better, if not comparable performances, with smaller parameter size, faster converging rate and better generalization.

2 Related Works

Reasoning in Discrete Space. Whether reasoning and planning are discrete or continuous by nature has always been an interesting question (Newell & Simon, 1976; Smolensky, 1988; Gershman et al., 2015; Garnelo & Shanahan, 2019; Mao et al., 2019). Most of the state-of-the-art methods are doing well in discrete reasoning tasks, for instance the Large Language Models (LLMs) doing various textual reasoning and planning tasks with chain-of-thoughts and various test-time decoding methods depending on this(Wei et al., 2023; Yao et al., 2023; Hu et al., 2024; Ma et al., 2024). Recently, using diffusion-based video models for planning in a continuous spatial-temporal space is also actively explored(Du et al., 2023; Feng et al., 2025; Yang et al., 2025). However, video models are prone to have drastic error-accumulation issues as the planning frames grow, and have to be assisted by classical search and pruning algorithms to accomplish long-horizon planning tasks, compromising considerable overheads for generationFeng et al. (2025); Yang et al. (2025). The similar issue also applies for other hierarchical planning methods modeling a continuous state dynamics with some vision-language models (VLMs) or vision-action models (VLAs)(Feng et al., 2025; Yang et al., 2025). At their center, there is commonly a high-level discrete planning kernel(Feng et al., 2025; Yang et al., 2025). Our work is an attempt to bridge the gap between the discrete planning and the continuous dynamics or controlling in a way of modeling the energy landscapes purely in a discrete space.

Discrete Diffusion Models. Recent work has extended score-based diffusion from continuous signals to categorical or tokenized domains. Austin et al. 2023 introduced forward processes that corrupt discrete tokens via uniform transition or mask schedules, then learn a neural reverse kernel parameterized by a transformer. The resulting training objective is a cross-entropy loss, yet yields a variational lower-bound analogous to continuous Evidence Lower Bound(ELBO). Compared with autoregressive baselines, discrete diffusion offers parallel sampling and explicit likelihood evaluation; relative to continuous-latent diffusion, it avoids decoder-induced blurriness and operates directly on interpretable symbols. Shi et al. 2025 further simplified the ELBO to a weighted integral of cross-entropy terms and allow state-dependent masking schedules, achieving strong likelihoods on text and images. Analogous to discrete diffusion, our work also utilize mask schedules in categorical domains. However, we discard the ELBO and take the more straightforward cross-entropy for training and Gibbs sampling for inference, in order to simplify the modeling of the annealed energy landscapes in the categorical domain.

Energy-Based Models. One of the major goal of machine learning is to model the dependency of the data. Energy-Based Model (EBM) achieves this by defining the dependency of an input $\boldsymbol{x}$ and an output $\boldsymbol{y}$ into a scalar $\mathcal{E}$, called energy. The optimal output $\boldsymbol{y}$ is achieved at the global minima of the energy landscape.

$$\arg\min_{\boldsymbol{y}} E(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R} \tag{1}$$

EBM has a statistical mechanics origin(Ackley et al., 1985), and in the context of probability theory and machine learning, it is usually combined with a Boltzmann distribution to represent a joint probability distribution:

$$\hat{\boldsymbol{y}} \sim p(\boldsymbol{x}, \boldsymbol{y}) = \frac{e^{-E(\boldsymbol{x}, \boldsymbol{y})}}{Z} \tag{2}$$

Alternatively, it can also be expressed in the form of scores by taking the gradients on both sides:

$$\nabla \log p \propto -\nabla E \tag{3}$$

In this sense, EBM can be connected with the diffusion model through a score-network, as they both features with forward and backward noise propagation guided by the model-generated gradients ∇E_θ or $\nabla \log p_\theta$ (Song & Ermon, 2020). However, most of the current EBMs or diffusion models are learned by their gradients, restricting them in a continuous context. In specific to reasoning and planning, Du et al. 2024 proposed IRED to learn a sequence of annealed EBMs to facilitate learning and sampling from the EBM landscape of the reasoning data. Our work adapted such annealed sequential design, and propose to learn the EBM from a discete perspective, which discards the ELBO and Langevin dynamics broadly used for training and sampling and instead applies the more simpler and gradient-agnostic cross entropy as the major training objective and Gibbs sampling for inference. LEAP is another highly related work proposed by Goyal et al. 2022, where the energy landscape of the planning trajectories is learned through a traditional MLM paradigm. LEAP also learns the EBM through a discrete objective and performs Gibbs sampling, but is domain-specific and can only inference action-by-action, which is less efficient and prone to accumulate errors.

3 DEBRA: A DISCRETE METHOD IN ENERGY-BASED REASONING

In this section, we formulate our training and inference paradigms, with an emphasis attached to the training part. We first set up our problem in 3.1. Then in 3.2, we introduce pseudolikelihood and demonstrate how we can derive the final form of cross-entropy loss aiming to maximize the pseudolikelihood of the desired tokens to be unmasked with the energy-based expression. To stabilize the training process across the annealing sequence of EBMs and to sharpen the landscapes, we further introduce a Hinge loss between each sample output and its corrupted negative version in 3.3 as an auxiliary contrastive objective for our training process. Finally, we briefly explains the Gibbs sampling we used for inference in 3.4. The final complete pseudocodes for training and inference are provided in Algorithm 1 and Algorithm 2.

3.1 PROBLEM SETTING

Assumption for Simplification. To initiate a tractable solution, we adopt the following assumption that intentionally simplify the core structure of the problem, deferring generalizations to later work: Given a labeled dataset $\mathcal{D} = \{X, Y\}$ for discrete reasoning task, we assume the input and output shares the same categorical space and both are of 2-dimensions: $(\boldsymbol{x}, \boldsymbol{y}) \in \mathcal{V}^{m \times d}$, where $\mathcal{V}$ is the vocabulary of size d and m is the total length of $\boldsymbol{x}$ and $\boldsymbol{y}$.[1] Such assumption allows us to waive

[1] In more practical settings, the input $\boldsymbol{x}$ and output $\boldsymbol{y}$ are usually of diverse modality and cannot trivially share the same categorical space, such as textual instruction being the input and the robot motion trajectories as the output. One possible way is to first embed them into a shared latent space. We plan to explore such generalization in our future work.

Algorithm 1 Debra Training

Input: Problem Dist $p_D(\boldsymbol{x}, \boldsymbol{y})$, Vocabulary $\mathcal{V}$, EBMs $\{E_\theta(\cdot\,; k)\}^K$, Landscapes K, Masking Schedule $\{m_k(\cdot)\}^K$, Corruption Function $c(\cdot)$, contrastive threshold τ.

while not converged **do**

▷ *Partially unmask to get the unobservables and observables:*

$\boldsymbol{x_u}, \boldsymbol{x_o} \leftarrow \boldsymbol{x} \oplus m_k(\boldsymbol{y}), \boldsymbol{x}, \boldsymbol{y} \sim p_D, k \sim K$

▷ *Use the k-th energy landscape to calculate the pseudolikelihood:*

for $i = 1$ to $|u|$: **do**

for $v = 1$ to $|\mathcal{V}|$: **do**

$\mathcal{E}_{i,v} \leftarrow E_\theta(x_{u_i} = v, \boldsymbol{x}_{\boldsymbol{u}_{-i}}, \boldsymbol{x_o}; k)$[a]

end for

$Z_i \leftarrow \frac{e^{-\mathcal{E}_i}}{\sum_{i'}^{|\mathcal{V}|} e^{-\mathcal{E}_{i'}}}$

end for

$\log p_{\text{pll}}(\boldsymbol{x_u}|\boldsymbol{x_o}) \leftarrow -|\boldsymbol{u}|\mathcal{E}^+ - \sum_{i=0}^{|\boldsymbol{u}|} \log Z_i$ (Ep.9)

$\mathcal{L}_{\text{CE}} \leftarrow \mathbb{E}_{p(\boldsymbol{x_u}|\boldsymbol{x_o})}[\log p_{\text{pll}}(\boldsymbol{x_u}|\boldsymbol{x_o})]$

▷ *Contrastively sharpen the energy landscapes:*

$\boldsymbol{x_u^-} \leftarrow c(\boldsymbol{x_u}, p_{\text{pll}}(\boldsymbol{x_u}|\boldsymbol{x_o}), \mathcal{E}^+)$

$\mathcal{E}^+ \leftarrow E_\theta(\boldsymbol{x_u}, \boldsymbol{x_o}; k)$

$\mathcal{E}^- \leftarrow E_\theta(\boldsymbol{x_u^-}, \boldsymbol{x_o}; k)$

$\mathcal{L}_{\text{Contrast}} \leftarrow \max(0, \tau + \mathcal{E}^+ - \mathcal{E}^-)$

▷ *Optimize objective $\mathcal{L}_{CE} + \mathcal{L}_{Contrast}$ wrt θ:*

$\Delta\theta \leftarrow \nabla_\theta(\mathcal{L}_{\text{CE}} + \mathcal{L}_{\text{Contrast}})$

Update θ based on $\Delta\theta$ using Adam optimizer

end while

[a] Here we denote $E_\theta(x_{u_i} = v \oplus \boldsymbol{x}_{u_{-i}} \oplus \boldsymbol{x_o}; k)$ as $E_\theta(x_{u_i} = v, \boldsymbol{x}_{u_{-i}}, \boldsymbol{x_o}; k)$ for readability. We continue with this simplified notation throughout this paper unless further specification.

the potential preprocessing of the inputs and outputs before calculating their corresponding energy $E(\boldsymbol{x}, \boldsymbol{y})$, and is directly applicable to the classical masked language modeling (MLM) setting.

We want to jointly learn a sequence of K annealed EBMs $\{E_\theta^k\}^{|K|}, k = \{1, ..., K\}$ in a scheduled discrete diffusion manner, where each EBM models the energy landscape of the observable and unobservable at that certain diffusion time stamp: $E_\theta(\boldsymbol{x}_o, \boldsymbol{x}_u; k)$:

$$\hat{\boldsymbol{x}_{\boldsymbol{u}}^k} = \arg\max \hat{p}(\hat{\boldsymbol{x}_{\boldsymbol{u}}^k}|\boldsymbol{x}_{\boldsymbol{o}}^k) \leftarrow E_\theta(\boldsymbol{x}_{\boldsymbol{o}}^k, \hat{\boldsymbol{x}_{\boldsymbol{u}}^k}; k) \tag{4}$$

Therefore, given the input $\mathbf{x}$ as the initial observable $\mathbf{x}_o$ and the fully masked output $[\text{MASK}]^{|\mathbf{y}|}$ as the initial unobservable $\mathbf{x}_{u;0}$, at each diffusion step k, the corresponding EBM E_θ^k guides to unmask part of the unobservables at the current step by minimizing the local energy, conditioned on the observables including the previously unmasked unobservables from previous steps. After K diffusion steps, we can fully recover the ultimate observable $\mathbf{x}_o \coloneqq \mathbf{x} \oplus \mathbf{y}$[2].

$$\begin{aligned}
\boldsymbol{x_o} &= \boldsymbol{x}, \\
\boldsymbol{x_u^0} &= [\text{MASK}]^{|\boldsymbol{y}|}, \\
\boldsymbol{x_o^k} &= \boldsymbol{x_o} \oplus \boldsymbol{x_u^{k-1}}, \qquad k = 1, \ldots, K \\
\boldsymbol{x_u^K} &= \boldsymbol{y}
\end{aligned} \tag{5}$$

[2] Here the $\oplus$ means integrating the observables and the unobservables while keeping their original order. For instance, in Sudoku, the unobservable positions to be unmasked are mixed with the observable inputs.

Algorithm 2 Debra Inference

Input: Input task $\boldsymbol{x}$, trained EBMs $\{E_\theta(\cdot\,; k)\}^K$, Landscapes K, Sampling Times T.
$\boldsymbol{x}_{\boldsymbol{u}}^{0,0} \leftarrow [\text{MASK}]^{|\boldsymbol{y}|}$
for each landscape $k = 1$ to K **do**
  $\boldsymbol{x}_{\boldsymbol{u}}^{k,0} \leftarrow \boldsymbol{x}_{\boldsymbol{u}}^{k-1,T}$
  ▷ *Gibbs sampling on candidate solution $\boldsymbol{x}_{\boldsymbol{u}}^{k}$ for T sampling times:*
  for run T times of sampling $t = 1$ to T **do**
    for $i = 1$ to $|u|$: **do**
      for $v = 1$ to $|\mathcal{V}|$: **do**
        $\mathcal{E}_{i,v}^{k,t} \leftarrow E_\theta(x'_{u_i} = v, \boldsymbol{x'}_{\boldsymbol{u}_{0:i-1}}^{k,t}, \boldsymbol{x'}_{\boldsymbol{u}_{i+1:|\boldsymbol{u}|}}^{k,t-1}, \boldsymbol{x}_{\boldsymbol{o}}; k)$
      end for
      $\hat{p_\theta}(x'^{k,t}_{u_i} | \boldsymbol{x}_o, \boldsymbol{x'}_{u_{0:i-1}}^{k,t}, \boldsymbol{x'}_{u_{i+1:|\boldsymbol{u}|}}^{k,t-1}) \leftarrow \frac{e^{-E_i^{k,t}}}{\sum_{i'}^{|\mathcal{V}|} e^{-E_{i'}^{k,t}}}$
      $x'^{k,t}_{u_i} \sim \hat{p_\theta}(x'^{k,t}_{u_i} | \boldsymbol{x}_o, \boldsymbol{x'}_{u_{0:i-1}}^{k,t}, \boldsymbol{x'}_{u_{i+1:|\boldsymbol{u}|}}^{k,t-1})$
    end for
    ▷ *Check if the sampled unmasked candidate decreases energy:*
    if $E_\theta(\boldsymbol{x'}_{\boldsymbol{u}}^{k,t}, \boldsymbol{x}_{\boldsymbol{o}}; k) < E_\theta(\boldsymbol{x}_{\boldsymbol{u}}^{k,t-1}, \boldsymbol{x}_{\boldsymbol{o}}; k)$ **then**
      $\boldsymbol{x}_{\boldsymbol{u}}^{k,t-1} \leftarrow \boldsymbol{x'}_{\boldsymbol{u}}^{k,t}$,
    end if
  end for
end for
return $\boldsymbol{y} = \boldsymbol{x}_{\boldsymbol{u}}$

An illustration of such scheduled unmasking procedure for inference is provided in Figure (). As for training, we simply perform the reverse: After K diffusion time stamps, the output $\mathbf{y}$ is fully masked so that the initial $\boldsymbol{x}_{\boldsymbol{o}} \oplus \boldsymbol{x}_{\boldsymbol{u}}^0 \coloneqq \boldsymbol{x} \oplus [\text{MASK}]^{|\boldsymbol{y}|}$ becomes the ultimate $\boldsymbol{x}_{\boldsymbol{o}} \oplus \boldsymbol{x}_{\boldsymbol{u}}^K \coloneqq \boldsymbol{x} \oplus \boldsymbol{y}$.

3.2 Cross-Entropy Loss with Pseudolikelihood

Inspired by Strauss & Oliva 2021, we adapt pseudolikelihood to approximate the likelihood of the unobservables conditioned on the observables for each step. Pseudolikelihood can approximate the joint probabilistic distribution with linear complexity. Say we want to approximate the probability of sequence $X \coloneqq X_0, X_1, \ldots, X_n \in \mathcal{V}^n$ with values $X_0 = x_0, X_1 = x_1, \ldots, X_n = x_n$ using pseudolikelihood:

$$p(x_0, x_1, \ldots, x_n) \approx \prod_{i=0}^{n} p(x_i | x_{j \neq i}) \coloneqq \prod_{i=0}^{n} p(x_i | \boldsymbol{x}_{-i}) \tag{6}$$

Therefore, we can reduce to calculation of the joint probability by composing a product of single-dimension conditionals(Strauss & Oliva, 2021)[3]. This is similar to the more commonly used autoregressive-style chain rule:

$$p(x_0, x_1, \ldots, x_n) = \prod_{i=0}^{n} p(x_i | \boldsymbol{x}_{0:i-1}) \tag{7}$$

However, the factors of pseudolikelihood can be computed simultaneously, and hence can be accelerated through vectorization and parallelization, which is adapted in our implementation.

Specifically in our setting, we want to approximate the conditional $p(\boldsymbol{x}_{\boldsymbol{u}} | \boldsymbol{x}_{\boldsymbol{o}})$ at each step k. Since the conditional $\boldsymbol{x}_{\boldsymbol{o}}$ is fixed, we can formulate the pseudolikelihood as follows:

[3] Strauss & Oliva 2021 didn't use the extact form of pseudolikelihood we formulated here, but instead used an autoregressive-style substitution for an order-agnostic conditional set problem. However, the main idea of breaking down the computation with single-dimension conditionals is similar and evident.

$$p(\boldsymbol{x}_u|\boldsymbol{x}_o) \approx \prod_{i=0}^{|\boldsymbol{u}|} p(x_{u_i}|\boldsymbol{x}_{\boldsymbol{u}_{-i}}; \boldsymbol{x}_{\boldsymbol{o}}) = \prod_{i=0}^{|\boldsymbol{u}|} p(x_{u_i}|\boldsymbol{x}_{\boldsymbol{o}} \oplus \boldsymbol{x}_{\boldsymbol{u}_{-i}}) \tag{8}$$

Then we substitute the factors with their energy-based formats (Eq.2), and finally we can derive the energy-based pseudo-log-likelihood as the final form (Appendix A):

$$\begin{aligned} p(\boldsymbol{x}_{\boldsymbol{u}}|\boldsymbol{x}_{\boldsymbol{o}}) &\approx \prod_{i=0}^{|\boldsymbol{u}|} p(x_{u_i}|\boldsymbol{x}_{\boldsymbol{u}_{-i}}; \boldsymbol{x}_{\boldsymbol{o}}) \\ &= \prod_{i=0}^{|\boldsymbol{u}|} \frac{e^{-E(x_{u_i}, \boldsymbol{x}_{\boldsymbol{u}_{-i}}, \boldsymbol{x}_{\boldsymbol{o}})}}{Z_i} \\ &\dots \\ \log p_{\text{pll}}(\boldsymbol{x}_{\boldsymbol{u}}|\boldsymbol{x}_{\boldsymbol{o}}) &\coloneqq -|\boldsymbol{u}|E(\boldsymbol{x}_{\boldsymbol{u}}, \boldsymbol{x}_{\boldsymbol{o}}) - \sum_{i=0}^{|\boldsymbol{u}|} \log Z_i \end{aligned} \tag{9}$$

Here the partition funciton $Z_i \coloneqq \sum_{x'_{u_i}}^{\mathcal{V}} e^{-E(x'_{u_i}, \boldsymbol{x}_{\boldsymbol{u}_{-i}}, \boldsymbol{x}_{\boldsymbol{o}})}$ can be calculated as a single-dimension categorical. For simplicity, we use brute force to calculate this partition. However, for larger vocabulary, we recommend using sampling methods, such as importance sampling to calculate this partition.

Finally, we use this pseudo-log-likelihood and the ground-truth conditional to calculate the cross entropy loss as our major training objective at each step:

$$\mathcal{L}_{\text{CE}} \coloneqq \mathbb{E}_{p(\boldsymbol{x}_{\boldsymbol{u}}|\boldsymbol{x}_{\boldsymbol{o}})}[\log p_{\text{pll}}(\boldsymbol{x}_{\boldsymbol{u}}|\boldsymbol{x}_{\boldsymbol{o}})] \tag{10}$$

3.3 Augmented Training with Contrastive Loss

Contrastive losses are commonly explored and adapted in many fields in machine learning, such as contrastive divergence(Hinton, 2002), max likelihood with Markov Chains(Hinton et al., 2006), adversarial generatorsGoodfellow et al. (2014), auto-encoders(Zhao et al., 2017; van den Oord et al., 2019), etc. In specific to the energy-based field, EBM methods can be generally classified as contrastive methods and regularized (architectural) methods, where the former pushing positive points up while pushing the negative ones down, and the latter limits the volume space that has lower energy. We design Debra to learn with a contrastive method, and adapt the Hinge loss specifically for each step k:

$$\mathcal{L}_{\text{contrast}} \coloneqq \max(0, \tau + E(\boldsymbol{x}_{\boldsymbol{o}}, \boldsymbol{x}_{\boldsymbol{u}}^{+}) - E(\boldsymbol{x}_{\boldsymbol{o}}, \boldsymbol{x}_{\boldsymbol{u}}^{-})) \tag{11}$$

Compared with other forms of contrastive losses, Hinge loss has an adjustable margin (i.e. the threshold τ) to define the contrastive degree, and it preserves the directional information of the energy landscape.

3.4 Gibbs Sampling for Flexible Test-Time Computation

As we learned a sequence of annealed EBMs, where each EBM models the energy landscape of the integration of observables and unobservables at that time step, we cam simply perform Gibbs sampling for T times iteratively during inference to uncover the unobservables. And after the last K-th EBM in the sequence completed T times of sampling, eventually we can recover the final prediction $\hat{\boldsymbol{y}}$:

Table 1: **Performance on Discrete Reasoning Tasks.** We compare Debra with other models on six mainstream discrete reasoning tasks: Countdown (CDn, where n means the number of operands, so n also indicates difficulty), Game of 24, Sudoku, Connectivity, and Binary Satisfiability

	Params	**CD3**	**CD4**	**CD5**	**Go24**	**Sudoku**	**Conn.**	**SAT**
Autoregressive								
GPT-2 Scratch	6M	94.1	31.9	4.3	-	-	-	-
	85M	95.9	45.8	5.1	18.8	20.1(?)	-	98(?)
LLaMA	7B	95.7	41.1	6.7	-	-	-	-
	13B	96.5	51.1	7.4	-	32.9	-	-
Diffusion								
D3PM	85M	99.4	83.1	27.6	-	-	-	-
MGDM	6M	-	-	-	-	100.0	-	-
	85M (?)	98.1	52.0	27.0	76.0	-	-	100(?)
Energy-Based Model								
IREM	?	-	-	-	-	93.5	94.3	-
IRED	?	-	-	-	-	99.4	99.1	-
Debra **(Ours)**	6M(BERT)	97.0	34.4(running)	-	-	(error?saturated)	-	-
	6M(GPT)	87.0	-	-	-	(error?saturated)	-	-

$$
\begin{aligned}
\boldsymbol{x}_{\boldsymbol{u}}^{0,0} &= [\text{MASK}]^{|\boldsymbol{y}|}, \\
&\dots, \\
\boldsymbol{x}_{\boldsymbol{u}}^{k,t} &= \{x_{u_i}^{k,t}\}_{i=0}^{|\boldsymbol{u}|}, x_{u_i}^{k,t} \sim p(x_{u_i}^{k,t} | \boldsymbol{x}_{u_{0:i-1}}^{k,t}, \boldsymbol{x}_{u_{i+1:|\boldsymbol{u}|}}^{k,t-1}), \qquad t = 0, \dots, T \\
&\dots, \\
\boldsymbol{x}_{\boldsymbol{u}}^{k+1,0} &= \boldsymbol{x}_{\boldsymbol{u}}^{k,T} \oplus [\text{MASK}]^{|\boldsymbol{u}_{k+1}|}, \qquad k = 0, \dots, K \\
&\dots, \\
\boldsymbol{x}_{\boldsymbol{u}}^{K,T} &= \hat{\boldsymbol{y}}
\end{aligned}
\tag{12}
$$

4 Experiments

To evaluate our paradigm, we test the performance of Debra in various discrete reasoning tasks, compared to various baseline models. In this section, we first introduce these tasks and baselines, and then offering our experiment settings and the results. We found that Debra achieves a superior balance of better or comparable performance, smaller and flexible parameterization, faster converging rate, and better generalizability. To examine the contribution of different components in our paradigm, we conduct ablation tests in aspects of masking scheme, base model, contrastive objective and sampling times. More experiment details can be found in Appendix().

4.1 Tasks

Countdown (Countdown, 2024) is a mathematical reasoning task where given a set of integer operands, a target integer number, arithmetic operators $(+, -, *, /)$ and comma ",", the goal is to connect the operators, comma and operands into multiple steps of equations to reach a target number. For instance, given input "15,44,79,50", we want to generate "44-15=29,79-29=50". We directly borrow the datasets collected from Ye et al. 2025, which contains three pairs of training and test sets corresponding for "three subtasks with increasing complexity by varying the number of input digits in 3,4,5".

Game of 24 is a specific version of Countdown (Ye et al., 2025). We train our models on Countdown 4 and evaluate the accuracy on the Game of 24 test set, as Yao et al. 2023 and Ye et al. 2025.

Sudoku (Park, 2016) is a classic constraint satisfaction problem. The player is required to fill a 9×9 board with some grids given the numbers. The goal is to fill all the blank grids such that every

row, column and 3×3 subgrids contains number 1 to 9. We also borrow the training and test sets from Ye et al. 2025. The samples are flattened into strings of 81 integer characters. For instance, the input is "080050060...603100007" (omitted for brevity), and the output is "789251364...653184297".

Connectivity is a graph connectivity task, where given an adjacency matrix, the goal is to predict the connectivity matrix. We adapt the training and test sets from Du et al. 2024.

Boolean Satisfiability (SAT) is another classic constraint satisfaction problem proven to be NP-complete (). Given a Boolean formula in the Conjunctive Normal Form (CNF), the goal is to decide whether a set of values assigned to the variables can make the formula evaluated to be True (1). We also borrowed the training and test sets from Ye et al. 2025 from three subtasks of increasing complexity.

4.2 Baselines

Here we introduce the baseline models we used for comparison. We didn't include the test-times methods such as using close-sourced large pretrained models utilizing CoT and search algorithms.

Auto-regressive models. While there is a wide range of auto-regressive models to choose from, we particularly pick the GPT-2 and LLaMA families from scratch for supervised fine-tuning (SFT). For open-source long-CoT Language Reasoning Models (LRMs), we choose Qwen3-8B and prompt them to generate the answer in the long-CoT thinking mode. For closed-source large pretrained models, we choose GPT5 and prompt them to generate the answer.

Discrete Diffusion models. As the highly related work from the discrete diffusion side, we choose the MGDM-tiny (6M) and MGDM-base (85M) (Ye et al., 2025) as our baseline.

Energy-based models. As the highly related work from the EBM side, we choose the IREM and the upgraded IRED (Du et al., 2022; 2024) as our baseline.

4.3 Experiment Settings

4.4 Results

4.5 Ablation tests

Masking Scheme.

Base Model.

Contrastive Objective.

Sampling Times.

5 Conclusions and Limitations

6 Acknowledgements

A Derivation of the Energy-Based Pseudo-log-likelihood

$$
\begin{aligned}
p(\boldsymbol{x}_u|\boldsymbol{x}_o) &\approx \prod_{i=0}^{|\boldsymbol{u}|} p(x_{u_i}|\boldsymbol{x}_{\boldsymbol{u}_{-i}};\boldsymbol{x}_{\boldsymbol{o}}) \\
\log p(\boldsymbol{x}_u|\boldsymbol{x}_o) &\approx \sum_{i=0}^{|\boldsymbol{u}|} \log p(x_{u_i}|\boldsymbol{x}_{\boldsymbol{u}_{-i}};\boldsymbol{x}_{\boldsymbol{o}}) \\
&= \sum_{i=0}^{|\boldsymbol{u}|} \log \frac{e^{-E(x_{u_i},\boldsymbol{x}_{\boldsymbol{u}_{-i}},\boldsymbol{x}_{\boldsymbol{o}})}}{\sum_{x'_{u_i}} e^{-E(x'_{u_i},\boldsymbol{x}_{\boldsymbol{u}_{-i}},\boldsymbol{x}_{\boldsymbol{o}})}} \\
&= \sum_{i=0}^{|\boldsymbol{u}|} (\log e^{-E(x_{u_i},\boldsymbol{x}_{\boldsymbol{u}_{-i}},\boldsymbol{x}_{\boldsymbol{o}})} - \log \sum_{x'_{u_i}} e^{-E(x'_{u_i},\boldsymbol{x}_{\boldsymbol{u}_{-i}},\boldsymbol{x}_{\boldsymbol{o}})}) \\
&= -\sum_{i=0}^{|\boldsymbol{u}|} E(x_{u_i},\boldsymbol{x}_{\boldsymbol{u}_{-i}},\boldsymbol{x}_{\boldsymbol{o}}) - \sum_{i=0}^{|\boldsymbol{u}|} \log \sum_{x'_{u_i}} e^{-E(x'_{u_i},\boldsymbol{x}_{\boldsymbol{u}_{-i}},\boldsymbol{x}_{\boldsymbol{o}})} \\
\log p_{\text{pll}}(\boldsymbol{x}_u|\boldsymbol{x}_o) &\coloneqq -|\boldsymbol{u}|E(\boldsymbol{x}_{\boldsymbol{u}},\boldsymbol{x}_{\boldsymbol{o}}) - \sum_{i=0}^{|\boldsymbol{u}|} \log Z_i
\end{aligned}
\tag{13}
$$

References

David H. Ackley, Geoffrey E. Hinton, and Terrence J. Sejnowski. A learning algorithm for boltzmann machines. *Cognitive Science*, 9(1):147–169, 1985. ISSN 0364-0213. doi: https://doi.org/10.1016/S0364-0213(85)80012-4. URL https://www.sciencedirect.com/science/article/pii/S0364021385800124.

Jacob Austin, Daniel D. Johnson, Jonathan Ho, Daniel Tarlow, and Rianne van den Berg. Structured denoising diffusion models in discrete state-spaces, 2023. URL https://arxiv.org/abs/2107.03006.

Anthony P. Chemero. *Radical Embodied Cognitive Science*. The MIT Press, 08 2009. ISBN 9780262258678. doi: 10.7551/mitpress/8367.001.0001. URL https://doi.org/10.7551/mitpress/8367.001.0001.

Hongyi Chen, Yilun Du, Yiye Chen, Joshua Tenenbaum, and Patricio A. Vela. Planning with sequence models through iterative energy minimization, 2023. URL https://arxiv.org/abs/2303.16189.

Countdown. Countdown (game show), 2024. URL https://en.wikipedia.org/wiki/Countdown_(game_show).

Yilun Du, Toru Lin, and Igor Mordatch. Model based planning with energy based models, 2021. URL https://arxiv.org/abs/1909.06878.

Yilun Du, Shuang Li, Joshua B. Tenenbaum, and Igor Mordatch. Learning iterative reasoning through energy minimization, 2022. URL https://arxiv.org/abs/2206.15448.

Yilun Du, Mengjiao Yang, Pete Florence, Fei Xia, Ayzaan Wahid, Brian Ichter, Pierre Sermanet, Tianhe Yu, Pieter Abbeel, Joshua B. Tenenbaum, Leslie Kaelbling, Andy Zeng, and Jonathan Tompson. Video language planning, 2023. URL https://arxiv.org/abs/2310.10625.

Yilun Du, Jiayuan Mao, and Joshua B. Tenenbaum. Learning iterative reasoning through energy diffusion, 2024. URL https://arxiv.org/abs/2406.11179.

Yunhai Feng, Jiaming Han, Zhuoran Yang, Xiangyu Yue, Sergey Levine, and Jianlan Luo. Reflective planning: Vision-language models for multi-stage long-horizon robotic manipulation, 2025. URL https://arxiv.org/abs/2502.16707.

Marta Garnelo and Murray Shanahan. Reconciling deep learning with symbolic artificial intelligence: representing objects and relations. *Current Opinion in Behavioral Sciences*, 29:17–23, 2019. ISSN 2352-1546. doi: https://doi.org/10.1016/j.cobeha.2018.12.010. URL https://www.sciencedirect.com/science/article/pii/S2352154618301943. Artificial Intelligence.

Samuel J. Gershman, Eric J. Horvitz, and Joshua B. Tenenbaum. Computational rationality: A converging paradigm for intelligence in brains, minds, and machines. *Science*, 349(6245):273–278, 2015. doi: 10.1126/science.aac6076. URL https://www.science.org/doi/abs/10.1126/science.aac6076.

Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014. URL https://arxiv.org/abs/1406.2661.

Kartik Goyal, Chris Dyer, and Taylor Berg-Kirkpatrick. Exposing the implicit energy networks behind masked language models via metropolis–hastings, 2022. URL https://arxiv.org/abs/2106.02736.

Geoffrey E. Hinton. Training products of experts by minimizing contrastive divergence. *Neural Computation*, 14(8):1771–1800, 08 2002. ISSN 0899-7667. doi: 10.1162/089976602760128018. URL https://doi.org/10.1162/089976602760128018.

Geoffrey E. Hinton, Simon Osindero, and Yee-Whye Teh. A fast learning algorithm for deep belief nets. *Neural Computation*, 18(7):1527–1554, 2006. doi: 10.1162/neco.2006.18.7.1527.

Zhiyuan Hu, Chumin Liu, Xidong Feng, Yilun Zhao, See-Kiong Ng, Anh Tuan Luu, Junxian He, Pang Wei Koh, and Bryan Hooi. Uncertainty of thoughts: Uncertainty-aware planning enhances information seeking in large language models, 2024. URL https://arxiv.org/abs/2402.03271.

O. Khatib. A unified approach for motion and force control of robot manipulators: The operational space formulation. *IEEE Journal on Robotics and Automation*, 3(1):43–53, 1987. doi: 10.1109/JRA.1987.1087068.

Yann Lecun, Sumit Chopra, and Raia Hadsell. *A tutorial on energy-based learning*. MIT Press, 01 2006.

Yunhao Luo, Chen Sun, Joshua B. Tenenbaum, and Yilun Du. Potential based diffusion motion planning, 2024. URL https://arxiv.org/abs/2407.06169.

Chang Ma, Haiteng Zhao, Junlei Zhang, Junxian He, and Lingpeng Kong. Non-myopic generation of language models for reasoning and planning, 2024. URL https://arxiv.org/abs/2410.17195.

Jiayuan Mao, Chuang Gan, Pushmeet Kohli, Joshua B. Tenenbaum, and Jiajun Wu. The neuro-symbolic concept learner: Interpreting scenes, words, and sentences from natural supervision, 2019. URL https://arxiv.org/abs/1904.12584.

John McCarthy and Patrick Hayes. Some philosophical problems from the standpoint of artificial intelligence. In B. Meltzer and Donald Michie (eds.), *Machine Intelligence 4*, pp. 463–502. Edinburgh University Press, 1969.

Allen Newell and Herbert A. Simon. Computer science as empirical inquiry: symbols and search. *Commun. ACM*, 19(3):113–126, March 1976. ISSN 0001-0782. doi: 10.1145/360018.360022. URL https://doi.org/10.1145/360018.360022.

Kyubyong Park. 1 million sudoku games, 2016. URL https://www.kaggle.com/bryanpark/sudoku.

Jiaxin Shi, Kehang Han, Zhe Wang, Arnaud Doucet, and Michalis K. Titsias. Simplified and generalized masked diffusion for discrete data, 2025. URL https://arxiv.org/abs/2406.04329.

Paul Smolensky. On the proper treatment of connectionism. *Behavioral and Brain Sciences*, 11(1): 1–23, 1988. doi: 10.1017/S0140525X00052432.

Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution, 2020. URL https://arxiv.org/abs/1907.05600.

Ryan R. Strauss and Junier B. Oliva. Arbitrary conditional distributions with energy, 2021. URL https://arxiv.org/abs/2102.04426.

A. M. Turing. On computable numbers, with an application to the entscheidungsproblem. *Proceedings of the London Mathematical Society*, s2-42(1):230–265, 01 1937. ISSN 0024-6115. doi: 10.1112/plms/s2-42.1.230. URL https://doi.org/10.1112/plms/s2-42.1.230.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding, 2019. URL https://arxiv.org/abs/1807.03748.

Lukas Vogelsang, Leila Drissi-Daoudi, and Michael Herzog. Processing load, and not stimulus evidence, determines the duration of unconscious visual feature integration. *Communications Psychology*, 1, 08 2023. doi: 10.1038/s44271-023-00011-2.

Haoyu Wang, Nan Wu, Hang Yang, Cong Hao, and Pan Li. Unsupervised learning for combinatorial optimization with principled objective relaxation, 2022. URL https://arxiv.org/abs/2207.05984.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models, 2023. URL https://arxiv.org/abs/2201.11903.

Yuncong Yang, Jiageng Liu, Zheyuan Zhang, Siyuan Zhou, Reuben Tan, Jianwei Yang, Yilun Du, and Chuang Gan. Mindjourney: Test-time scaling with world models for spatial reasoning, 2025. URL https://arxiv.org/abs/2507.12508.

Zhutian Yang, Jiayuan Mao, Yilun Du, Jiajun Wu, Joshua B. Tenenbaum, Tomás Lozano-Pérez, and Leslie Pack Kaelbling. Compositional diffusion-based continuous constraint solvers, 2023. URL https://arxiv.org/abs/2309.00966.

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. URL https://arxiv.org/abs/2305.10601.

Jiacheng Ye, Jiahui Gao, Shansan Gong, Lin Zheng, Xin Jiang, Zhenguo Li, and Lingpeng Kong. Beyond autoregression: Discrete diffusion for complex reasoning and planning, 2025. URL https://arxiv.org/abs/2410.14157.

Junbo Zhao, Michael Mathieu, and Yann LeCun. Energy-based generative adversarial network, 2017. URL https://arxiv.org/abs/1609.03126.

Lei Zhuang, Jingdong Zhao, Yuntao Li, Zichun Xu, Liangliang Zhao, and Hong Liu. Transformer-enhanced motion planner: Attention-guided sampling for state-specific decision making. *IEEE Robotics and Automation Letters*, 9(10):8794–8801, October 2024. ISSN 2377-3774. doi: 10.1109/lra.2024.3450305. URL http://dx.doi.org/10.1109/LRA.2024.3450305.