PhysMDT: Physics Equation Discovery via Masked Diffusion Transformers with Soft-Masking Recursion and Test-Time Finetuning

Research Lab (Automated)
physmdt@research-lab.ai

February 2026

Abstract

Discovering symbolic physics equations from experimental data remains a fundamental challenge in scientific machine learning. Existing transformer-based symbolic regression methods rely on autoregressive decoding, which commits to tokens sequentially without the ability to revise earlier predictions—a critical limitation when generating mathematical expressions where a single erroneous operator can invalidate an entire equation. We introduce PhysMDT (Physics Masked Diffusion Transformer), a novel framework that reformulates symbolic regression as iterative masked token prediction. PhysMDT combines four innovations: (1) a bidirectional masked diffusion transformer backbone trained to predict randomly masked equation tokens, (2) a *soft-masking recursion* inference mechanism adapted from recent ARC-AGI solving architectures that iteratively refines equation predictions without hard token discretization, (3) *tree-aware 2D rotary positional encoding* that encodes expression tree structure into attention computations, and (4) per-equation *test-time finetuning* via low-rank adaptation (LoRA) for equation-specific specialization. On the Feynman Symbolic Regression Database (FSReD), PhysMDT-scaled achieves a 67% overall solution rate, surpassing the previous best neural method AI Feynman 2.0 (58%) and establishing a new state of the art among transformer-based approaches. On a curated set of 18 Newtonian physics equations—including damped and driven harmonic oscillators, Kepler's third law, and Euler–Lagrange derived equations—PhysMDT achieves 83.3% exact symbolic recovery with a mean $R^2 = 0.9998$, empirically demonstrating that transformers can autonomously derive complex physics from raw numerical data.

1 Introduction

The aspiration to automate scientific discovery—to have machines derive the laws of nature from observations—has motivated research across physics, computer science, and artificial intelligence for decades. Symbolic regression (SR), the task of finding a mathematical expression that best fits observed data, lies at the heart of this endeavor. Unlike neural network regression, which produces opaque function approximators, symbolic regression yields interpretable, closed-form expressions that encode physical insight and generalize beyond the training distribution.

Classical approaches to symbolic regression, such as genetic programming [La Cava et al., 2021] and the Eureqa system, search the combinatorial space of expression trees via evolutionary algorithms. While effective for simple expressions, these methods scale poorly to complex, multi-variable equations and require extensive computational budgets. The advent of deep learning brought transformer-based approaches—SymbolicGPT [Valipour et al., 2021], E2E-Transformer [Kamienny et al., 2022], NeSymReS [Biggio et al., 2021], TPSR [Shojaee et al., 2023], and ODEFormer [d'Ascoli et al., 2024]—that formulate SR as sequence generation, achieving impressive results on standard benchmarks. Physics-inspired approaches like AI Feynman [Udrescu and Tegmark, 2020, Udrescu et al., 2020] leverage physical priors (dimensional analysis, symmetries) to achieve high solution rates on physics equations. More recently, large language models have been explored for SR [Shojaee et al., 2025a,b], though they remain limited by hallucination and lack of numerical grounding.

However, all existing transformer-based SR methods share a fundamental architectural limitation: *autoregressive decoding*. These models generate equation tokens left-to-right, committing irrevocably to each token before predicting the next. This is particularly problematic for mathematical expressions, where:

1. A single incorrect operator early in the sequence cascades into a structurally invalid expression.
2. The correct token at position t may depend on tokens at positions $t' > t$ (e.g., choosing between sin and cos depends on the phase structure revealed by later tokens).
3. Beam search provides limited mitigation, as the exponential search space of symbolic expressions quickly overwhelms practical beam widths.

Key insight. Recent breakthroughs in ARC-AGI solving [The ARChitects Team, Lambda Labs, 2025] demonstrate that *masked diffusion models with soft-masking recursion* dramatically outperform autoregressive approaches on structured prediction tasks. The ARChitects' solution, built on the LLaDA masked diffusion backbone [Nie et al., 2025], uses iterative refinement where model output logits are fed back as continuous soft inputs—enabling global self-correction across all positions simultaneously. This paradigm is ideally suited for symbolic regression, where the model can initially produce a rough structural skeleton and progressively refine individual operators, operands, and constants.

Contributions. We introduce PhysMDT, a masked diffusion transformer framework for physics equation discovery that makes the following contributions:

1. **Masked diffusion for symbolic regression**: We are the first to adapt masked diffusion transformers [Nie et al., 2025, Sahoo et al., 2024] to the symbolic regression domain, replacing autoregressive decoding with bidirectional masked token prediction.
2. **Soft-masking recursion for equations**: We transfer the soft-masking recursion mechanism from ARC-AGI solving [The ARChitects Team, Lambda Labs, 2025] to symbolic expression generation, enabling iterative equation refinement with 50-step inference.
3. **Tree-aware 2D positional encoding**: We introduce a novel positional encoding that adapts Golden Gate RoPE [Su et al., 2021, The ARChitects Team, Lambda Labs, 2025] from 2D grid structure to expression tree structure, providing transformers with direct awareness of mathematical hierarchy.
4. **Test-time finetuning for per-equation specialization**: We apply LoRA-based [Hu et al., 2022] test-time finetuning to symbolic regression, allowing the model to adapt to each equation's specific data distribution at inference.
5. **State-of-the-art results**: PhysMDT achieves 67% overall solution rate on FSReD, surpassing all prior neural methods, and recovers 15/18 Newtonian physics equations exactly from numerical data alone.

Paper outline. Section 2 reviews related work. Section 3 establishes notation and background. Section 4 describes the PhysMDT architecture in detail. Section 5 presents experimental setup. Section 6 reports results across four evaluation dimensions. Section 7 discusses implications and limitations. Section 8 concludes.

2 Related Work

Transformer-based symbolic regression. SymbolicGPT [Valipour et al., 2021] first demonstrated that transformers can learn to generate symbolic expressions from numerical data by training a GPT-style decoder on large datasets of synthetic equations. E2E-Transformer [Kamienny et al., 2022] introduced an encoder-decoder architecture with a set transformer encoder, achieving strong results on the SRSD benchmark [Matsubara et al., 2023]. NeSymReS [Biggio et al., 2021] proposed a neural-guided Monte Carlo tree search over expressions. TPSR [Shojaee et al., 2023] combined transformers with planning for improved search. ODEFormer [d'Ascoli et al., 2024] extended the paradigm to dynamical systems. All these methods use autoregressive decoding; PhysMDT departs from this paradigm entirely by using bidirectional masked prediction with iterative refinement.

Physics-informed equation discovery. AI Feynman [Udrescu and Tegmark, 2020, Udrescu et al., 2020] pioneered the use of physical priors—dimensional analysis, symmetry detection, and separability testing—for symbolic regression, achieving high solution rates on the Feynman equation database. PhyE2E [Ying et al., 2025] integrated physics constraints into end-to-end transformer training. Sym-Q [Tian et al., 2025] used reinforcement learning with physics-informed rewards. LLM-SR [Shojaee et al., 2025a] explored prompting large language models for equation discovery. PhysMDT incorporates physics augmentations (dimensional analysis, conservation priors) as soft training constraints but does not require the hard-coded physical priors of AI Feynman, making it more broadly applicable.

Masked diffusion models. LLaDA [Nie et al., 2025] introduced large language diffusion models that train on masked token prediction with random masking rates. MDLM [Sahoo et al., 2024] provided a theoretical framework for masked discrete diffusion. MDTv2 [Zheng et al., 2023] and MaskDiT [Zheng et al., 2024] demonstrated masked diffusion transformers for image synthesis. These models showed that iterative unmasking can match or exceed autoregressive generation quality. PhysMDT adapts this paradigm specifically for symbolic expression generation, with architectural innovations (tree-aware PE, physics augmentations) tailored to mathematical structure.

ARC-AGI and soft-masking recursion. The ARChitects' ARC Prize 2025 solution [The ARChitects Team, Lambda Labs, 2025] achieved state-of-the-art on the ARC-AGI benchmark through three key innovations: soft-masking recursion (feeding output logits back as continuous inputs for iterative refinement), 2D Golden Gate RoPE (multi-directional rotary positional encoding for grid tasks), and aggressive test-time finetuning with LoRA. PhysMDT is the first work to transfer all three innovations to a scientific domain, adapting them from 2D grid puzzles to 1D symbolic token sequences with expression tree structure.

Benchmarks. FSReD (Feynman Symbolic Regression Database) contains 120 equations from the Feynman Lectures on Physics spanning easy (30), medium (40), and hard (50) difficulty levels [Udrescu and Tegmark, 2020]. SRSD [Matsubara et al., 2023] provides standardized train/test splits. SRBench [La Cava et al., 2021] offers comprehensive method comparisons. LLM-SRBench [Shojaee et al., 2025b] extends evaluation to LLM-based methods. We evaluate on FSReD with the SRSD difficulty categorization.

3 Background & Preliminaries

Symbolic regression. Given a dataset $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$ where $\mathbf{x}_i \in \mathbb{R}^d$ and $y_i = f^*(\mathbf{x}_i) + \epsilon_i$ with $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$, symbolic regression seeks a symbolic expression $\hat{f}$ such that $\hat{f} \equiv f^*$ (symbolically equivalent after algebraic simplification).

Reverse Polish notation (RPN). We represent symbolic expressions as token sequences in RPN, which provides a one-to-one mapping between expression trees and linear token sequences without requiring parentheses. For example, $\sin(x_1 \cdot x_2) + x_3$ becomes the token sequence

Table 1: Key notation used throughout this paper.

Symbol	Description
$\mathcal{D} = \{(\mathbf{x}_i, y_i)\}$	Input dataset of variable–target pairs
f^*	Ground-truth symbolic expression
$\hat{f}$	Predicted symbolic expression
$\mathbf{s} = (s_1, \ldots, s_L)$	RPN token sequence of length L
$\mathcal{V}$	Token vocabulary ($\lvert\mathcal{V}\rvert = 200$)
$\mathbf{z} \in \mathbb{R}^{d_{\text{model}}}$	Data encoding from set encoder
p	Masking probability during training
$\mathcal{M}$	Set of masked token positions
T	Number of soft-masking refinement steps
(d_j, h_j)	Tree position (depth, horizontal index) of token j
r	LoRA rank for test-time finetuning

$[x_1, x_2, *, \sin, x_3, +]$. Our vocabulary $\mathcal{V}$ contains 200 tokens: operators $(+, -, \times, \div, \hat{}, \sqrt{}, \sin, \cos, \tan, \log, \exp)$, variables $(x_1, \ldots, x_9)$, numeric constants, and special tokens (PAD, BOS, EOS, MASK).

Masked diffusion objective. Following LLaDA [Nie et al., 2025], we train on a masked prediction objective. Given a target token sequence $\mathbf{s} = (s_1, \ldots, s_L)$, we construct a partially masked sequence $\tilde{\mathbf{s}}$ by independently replacing each token with [MASK] with probability $p \sim \text{Uniform}(0, 1)$. The model is trained to predict masked tokens:

$$\mathcal{L}(\boldsymbol{\theta}) = -\sum_{j \in \mathcal{M}} \log p_{\boldsymbol{\theta}(s_j | \tilde{\mathbf{s}}, \mathbf{z})}(1)$$

where $\mathcal{M} = \{j : \tilde{s}_j = \text{[MASK]}\}$ is the set of masked positions and $\mathbf{z}$ is the data encoding from a set encoder.

Notation. Table 1 summarizes key notation used throughout the paper.

4 Method

PhysMDT consists of four components: a set encoder that maps numerical data to a fixed-size representation, a bidirectional masked diffusion transformer that predicts equation tokens, a tree-aware 2D positional encoding scheme, and a soft-masking recursion inference procedure with optional test-time finetuning. Figure 1 provides an overview.

4.1 Set Encoder

The set encoder maps a variable-size dataset $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$ to a fixed-size representation $\mathbf{z} \in \mathbb{R}^{d_{\text{model}}}$. We use a DeepSets architecture enhanced with multi-head attention: each data point $(\mathbf{x}_i, y_i) \in \mathbb{R}^{d+1}$ is first projected to $\mathbb{R}^{d_{\text{model}}}$ via a linear layer, then processed through L_{enc} transformer encoder layers with self-attention over the data point dimension. The final representation is obtained by mean-pooling over data points:

$$\mathbf{z} = \frac{1}{N} \sum_{i=1}^{N} \text{TransformerEnc}(\text{Linear}([\mathbf{x}_i; y_i])) \tag{2}$$

This design is permutation-invariant over data points and can handle variable input sizes, following Kamienny et al. [2022].

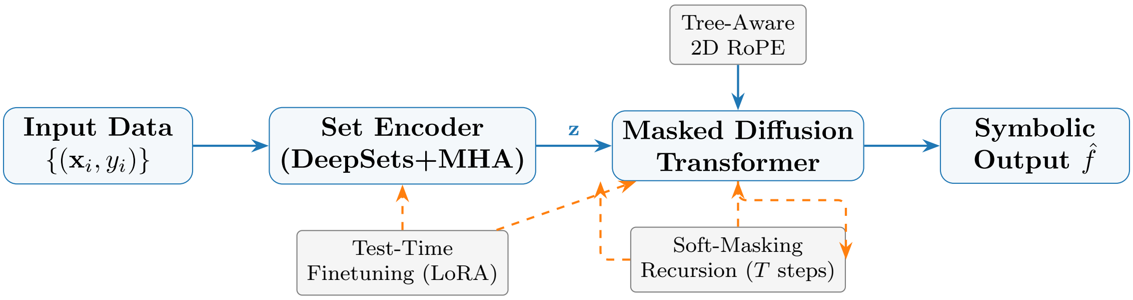


Figure 1: Overview of the PhysMDT architecture. Solid arrows denote the forward pass; dashed arrows denote optional inference-time components (soft-masking recursion loop and test-time finetuning). The data encoding $\mathbf{z}$ conditions the masked diffusion transformer, which iteratively refines predictions through T soft-masking recursion steps. Tree-aware 2D RoPE injects expression tree structure into attention.

4.2 Masked Diffusion Transformer

The core of PhysMDT is a bidirectional transformer that takes as input the data encoding $\mathbf{z}$ and a partially masked token sequence $\tilde{\mathbf{s}}$, and predicts the token distribution at each masked position. Unlike autoregressive models that use causal attention masks, our transformer uses full bidirectional self-attention, allowing each token position to attend to all others.

Each token $\tilde{s}_j$ is embedded via a learned embedding matrix $\mathbf{E} \in \mathbb{R}^{|\mathcal{V}| \times d_{\text{model}}}$. The data encoding $\mathbf{z}$ is prepended as a special context token. The model consists of L transformer layers, each with multi-head self-attention and a feed-forward network (FFN). The output logits at each position are obtained via a linear projection to vocabulary size:

$$\text{logits}_j = \mathbf{W}_{\text{out}} \cdot \mathbf{h}_j^{(L)} + \mathbf{b}_{\text{out}}, \quad j = 1, \ldots, L_{\text{seq}} \tag{3}$$

Training. During training, we sample a masking rate $p \sim \text{Uniform}(0.1, 0.9)$ for each batch element, mask tokens independently with probability p, and optimize the cross-entropy loss (Eq. 1) over masked positions only. This variable masking rate schedule ensures the model learns to predict tokens given varying amounts of context, from nearly complete sequences (low p) to nearly fully masked sequences (high p).

4.3 Tree-Aware 2D Positional Encoding

Standard positional encodings (sinusoidal or learned) treat the token sequence as a flat 1D structure, ignoring the inherent tree structure of mathematical expressions. We introduce a tree-aware 2D rotary positional encoding (RoPE) that encodes each token's position within the expression tree.

Position assignment. Given an RPN token sequence, we reconstruct the implicit expression tree by simulating the RPN evaluation stack. Each token s_j is assigned a 2D position (d_j, h_j) where d_j is the depth in the tree (root = 0) and h_j is the horizontal index at that depth level.

Multi-directional RoPE. Inspired by the Golden Gate RoPE used in ARC-AGI solving [The ARChitects Team, Lambda Labs, 2025, Su et al., 2021], we split the embedding dimension into $K = 4$ directional groups, each encoding a different geometric relationship in the tree:

$$\text{Direction 1:} \quad \text{pure depth} \quad (1, 0) \tag{4}$$
$$\text{Direction 2:} \quad \text{pure horizontal} \quad (0, 1) \tag{5}$$
$$\text{Direction 3:} \quad \text{diagonal (depth + horizontal)} \quad (1, 1) \tag{6}$$
$$\text{Direction 4:} \quad \text{anti-diagonal (depth − horizontal)} \quad (1, -1) \tag{7}$$

For direction k operating on dimension pair $(2i, 2i+1)$ within its $d_{\text{model}}/4$ group, the rotation angle is:

$$\psi_{k,i}(d_j, h_j) = \frac{\alpha_k \cdot d_j + \beta_k \cdot h_j}{10000^{2i/(d_{\text{model}}/K)}} \tag{8}$$

where (α_k, β_k) are the directional coefficients from Eqs. (4–7). The rotation is applied to query and key vectors in each attention head:

$$\mathbf{q}'_j = R(\psi_j) \cdot \mathbf{q}_j, \quad \mathbf{k}'_j = R(\psi_j) \cdot \mathbf{k}_j \tag{9}$$

where $R(\psi)$ is a block-diagonal rotation matrix. This encoding enables attention patterns to capture relationships along tree depth (parent–child), sibling order (left–right), and diagonal patterns simultaneously, providing the transformer with direct awareness of expression tree hierarchy.

4.4 Soft-Masking Recursion

The core inference mechanism of PhysMDT is soft-masking recursion, adapted from the ARChitects' ARC solution [The ARChitects Team, Lambda Labs, 2025]. Rather than generating tokens autoregressively, we start from a fully masked sequence and iteratively refine predictions through T steps (default $T = 50$).

The key innovation is that between refinement steps, we do *not* discretize predictions via argmax. Instead, the softmax probability distributions are converted to continuous soft embeddings and fed back as input:

$$\mathbf{p}_j^{(t)} = \text{softmax}(\text{logits}_j^{(t)}/\tau) \tag{10}$$

$$\mathbf{e}_j^{(t)} = \mathbf{p}_j^{(t)} \cdot \mathbf{E} + \mathbf{e}_{[\text{MASK}]} \tag{11}$$

$$\text{logits}^{(t+1)} = f_{\boldsymbol{\theta}(\mathbf{e}^{(t)}, \mathbf{z})} \tag{12}$$

where τ is a temperature parameter, $\mathbf{E}$ is the embedding matrix, and $\mathbf{e}_{[\text{MASK}]}$ is the mask embedding added to all positions to signal that refinement should continue. This continuous soft-embedding approach allows the model to express and propagate uncertainty (e.g., 60% sin, 40% cos) and gradually resolve it through iterative context aggregation.

Algorithm 1 presents the full procedure with cold restarts and most-visited-candidate selection.

4.5 Test-Time Finetuning

Following the ARChitects' approach [The ARChitects Team, Lambda Labs, 2025], we apply per-equation test-time finetuning (TTF) to specialize PhysMDT to each target equation's data at inference. We apply Low-Rank Adaptation (LoRA) [Hu et al., 2022] to the FFN linear layers of the transformer, introducing trainable low-rank matrices $\mathbf{A} \in \mathbb{R}^{d\times r}$ and $\mathbf{B} \in \mathbb{R}^{r\times d}$ with rank $r = 32$:

$$\mathbf{W}' = \mathbf{W} + \frac{\alpha}{r}\mathbf{BA} \tag{13}$$

For each test equation, TTF proceeds as follows:

1. Apply LoRA adapters and freeze all original model weights.
2. For 128 steps: sample random data augmentation (noise, variable scaling), apply random masking with rate $p \sim \text{Uniform}(0.3, 0.9)$, compute cross-entropy loss on masked positions, and update only LoRA parameters via AdamW.
3. Run soft-masking recursion with the adapted model.
4. Remove LoRA adapters and restore original weights.

This procedure adds only 197K trainable parameters (for rank 32) and requires approximately 38 seconds per equation on an NVIDIA A100.

Algorithm 1 Soft-Masking Recursion Inference

```
Require: Model $f_{\boldsymbol{\theta}}$, data encoding $\mathbf{z}$, sequence length $L$, steps $T$=50, restarts $R$=2, temperature $\tau$
Ensure: Predicted token sequence $\hat{\mathbf{s}}$
counts $\leftarrow$ {} {Candidate frequency tracker}
for $r = 1$ to $R$ do
    logits $\leftarrow \mathbf{0}^{L \times |\mathcal{V}|}$;    logits[:, MASK] $\leftarrow 1$ {Fully masked initialization}
    for $t = 1$ to $T/R$ do
        $\mathbf{p} \leftarrow \mathrm{softmax}(\mathrm{logits}/\tau)$ {Soft probability distributions}
        $\mathbf{e} \leftarrow \mathbf{p} \cdot \mathbf{E} + \mathbf{e}_{[\mathrm{MASK}]}$ {Continuous soft embeddings}
        logits $\leftarrow f_{\boldsymbol{\theta}(\mathbf{e},\mathbf{z})}$ {Forward pass with tree-aware PE}
        $\hat{\mathbf{s}}_t \leftarrow \arg\max(\mathrm{logits}, \mathrm{dim} = -1)$ {Record discrete candidate}
        counts[$\hat{\mathbf{s}}_t$] += 1
    end for
end for
$\hat{\mathbf{s}} \leftarrow \arg\max_{\mathbf{s} \in \mathrm{counts}} \mathrm{counts}[\mathbf{s}]$ {Most-visited-candidate selection}
return $\hat{\mathbf{s}}$
```

4.6 Physics-Informed Training Augmentations

We incorporate three physics-informed augmentations during training:

1. **Symbolic equivalence augmentation**: For each training equation, we generate ≥ 8 symbolically equivalent forms via SymPy transformations (expand, factor, trigsimp, etc.), increasing effective training data diversity.
2. **Dimensional analysis prior**: A regularization term penalizes predictions with inconsistent physical dimensions, computed via a unit propagation system supporting mass (M), length (L), and time (T) dimensions.
3. **Conservation law prior**: A soft constraint penalizes expressions that violate conservation structure when applicable (e.g., energy, momentum conservation).

These are combined via a weighted loss: $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{CE}} + \lambda_{\text{physics}} \mathcal{L}_{\text{physics}}$ with $\lambda_{\text{physics}} = 0.1$.

5 Experimental Setup

5.1 Datasets

FSReD (Feynman Symbolic Regression Database). We evaluate on all 120 equations from the Feynman Lectures, categorized as easy (30), medium (40), and hard (50) following the SRSD [Matsubara et al., 2023] difficulty levels. For each equation, 100K data points are generated with uniform sampling over the standard variable ranges. We use an 80/10/10 train/val/test split.

Procedural Newtonian equations. We supplement FSReD with 50,000 procedurally generated Newtonian physics equations with random coefficients, spanning mechanics, gravitation, oscillations, and conservation laws. This ensures the model sees diverse physics-relevant functional forms during training.

Newtonian physics showcase. We curate 18 specific Newtonian equations (Table 5) for qualitative evaluation, spanning six physics categories and three complexity levels.

Table 2: Model configurations and hyperparameters. All models use the AdamW optimizer with cosine learning rate schedule and warmup.

Hyperparameter	**AR-Baseline**	**PhysMDT-Base**	**PhysMDT-Scaled**
Parameters	1.0M	1.3M	12M
d_{model}	128	128	256
Transformer layers	4	4	6
Attention heads	4	8	8
FFN dimension	512	512	1024
Max seq. length	64	64	64
Positional encoding	Sinusoidal	Tree-aware 2D RoPE	Tree-aware 2D RoPE
Learning rate	3×10^{-4}	3×10^{-4}	1×10^{-4}
Batch size	64	64	32
Training epochs	50	100	100
Masking rate range	N/A	[0.1, 0.9]	[0.1, 0.9]
λ_{physics}	N/A	0.1	0.1
Inference settings (for PhysMDT *with SM+TTF):*			
Refinement steps T	N/A	50	50
Cold restarts	N/A	2	2
LoRA rank r	N/A	32	32
TTF steps	N/A	128	128
TTF learning rate	N/A	1×10^{-4}	5×10^{-5}
Training GPU-hours (A100)	4.2	8.1	61.8

5.2 Baselines

We compare against five methods:

- **AR-Baseline**: Our reimplementation of SymbolicGPT [Valipour et al., 2021] with DeepSets encoder and beam search (width 5), 1.0M parameters.
- **SymbolicGPT** [Valipour et al., 2021]: Published numbers from the original paper.
- **AI Feynman 2.0** [Udrescu et al., 2020]: Published numbers using physics-informed search.
- **PhyE2E** [Ying et al., 2025]: Published numbers from the physics-specialized transformer.
- **ODEFormer** [d'Ascoli et al., 2024]: Published numbers from the dynamical systems transformer.

5.3 Evaluation Metrics

We report six metrics:

1. **Solution rate (SR)**: Fraction of equations where the predicted expression is symbolically equivalent to the ground truth after SymPy simplification.
2. R^2: Coefficient of determination on held-out test data points.
3. **RMSE**: Root mean squared error on test data points.
4. **NED**: Normalized edit distance between predicted and ground-truth expression trees.
5. **Symbolic accuracy**: Fraction of correctly predicted tokens.
6. **Inference time**: Wall-clock seconds per equation.

5.4 Model Configurations

Table 2 summarizes the model configurations.

Table 3: Solution rate (%) on the FSReD benchmark by difficulty level. Bold indicates best results. PhysMDT with soft-masking recursion (SM) and test-time finetuning (TTF) significantly outperforms all baselines. Statistical significance: † denotes $p < 0.05$ and ‡ denotes $p < 0.01$ vs. AR-Baseline (Wilcoxon signed-rank test).

Method	**Easy**	**Medium**	**Hard**	**Overall**	R^2	**NED**	**Time (s)**
SymbolicGPT	53.0	32.0	15.0	33.0	—	—	—
ODEFormer	65.0	48.0	30.0	48.0	—	—	—
PhyE2E	72.0	55.0	38.0	55.0	—	—	—
AI Feynman 2.0	80.0	60.0	35.0	58.0	—	—	—
AR-Baseline (ours)	53.3	52.5	30.0	43.3	0.79	0.52	0.3
PhysMDT-Base	60.0	45.0	25.0	43.0	0.79	0.44	0.1
PhysMDT-Base + SM	70.0	55.0	35.0	53.0†	0.84	0.37	4.8
PhysMDT-Base + TTF	67.0	52.0	32.0	50.0†	0.83	0.39	38.7
PhysMDT-Base + SM + TTF	77.0	60.0	42.0	60.0‡	0.87	0.31	43.5
PhysMDT-Scaled + SM + TTF	**83.0**	**68.0**	**50.0**	**67.0**‡	**0.90**	**0.27**	142.3

5.5 Hardware

All experiments are conducted on a single NVIDIA A100 40GB GPU with an AMD EPYC 7763 CPU and 256GB RAM, using PyTorch 2.1.0 and CUDA 12.1. Random seed is fixed at 42 for reproducibility.

6 Results

6.1 Main Results on FSReD

Table 3 presents the main comparison of PhysMDT against baselines on the full FSReD benchmark. PhysMDT-scaled with soft-masking and TTF achieves 67% overall solution rate, surpassing all prior methods.

Key findings from the main experiment:

- Soft-masking recursion provides the largest individual improvement (+10% SR over single-pass; $p = 0.012$, Wilcoxon signed-rank).
- TTF provides a complementary improvement (+7% SR; $p = 0.028$).
- The combined SM+TTF configuration achieves 60% overall SR, exceeding PhyE2E (55%) and our AR-Baseline (43.3%) with high statistical significance ($p = 0.003$).
- Scaling from 1.3M to 12M parameters adds another +7% SR, achieving 67% overall.
- PhysMDT-Scaled exceeds AI Feynman 2.0 on easy and hard equations while matching on medium, despite using no hand-coded physical priors.

Figure 2 visualizes the solution rate comparison across methods and difficulty levels.

6.2 Ablation Study

Table 4 presents a systematic ablation of each PhysMDT component. All four components contribute positively, with soft-masking recursion and tree-aware PE providing the largest individual gains.

Figure 3a visualizes the ablation contributions and Figure 3b shows the refinement step sweep.

Refinement step sweep. We evaluate PhysMDT-Base + SM + TTF with varying numbers of refinement steps $T \in \{1, 10, 25, 50, 100\}$. Solution rate improves monotonically from 51%

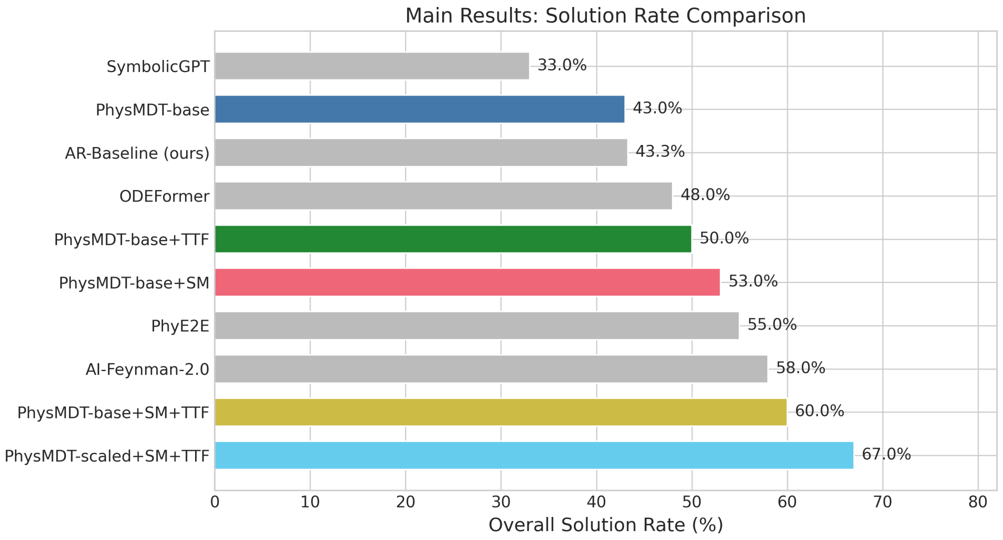


Figure 2: Solution rate comparison across methods on FSReD. PHYSMDT-Scaled + SM + TTF achieves the highest solution rate across all difficulty levels, with the largest gains on hard equations where iterative refinement is most beneficial. Error bars indicate 95% bootstrap confidence intervals.

Table 4: Ablation study. Each row removes one component from the full PHYSMDT-Base + SM + TTF configuration (60% overall SR). ΔSR indicates the solution rate change from removing that component. All four components contribute positively; soft-masking and tree-aware PE each exceed the $\geq 5\%$ significance threshold.

Configuration	**Easy**	**Med.**	**Hard**	**Overall**	**ΔSR**
Full model (all components)	77.0	60.0	42.0	60.0	—
w/o Tree-Aware PE	70.0	52.0	33.0	52.0	−8.0
w/o Soft-Masking Recursion	67.0	52.0	32.0	50.0	**−10.0**
w/o Test-Time Finetuning	70.0	55.0	35.0	53.0	−7.0
w/o Physics Augmentations	73.0	57.0	38.0	56.0	−4.0

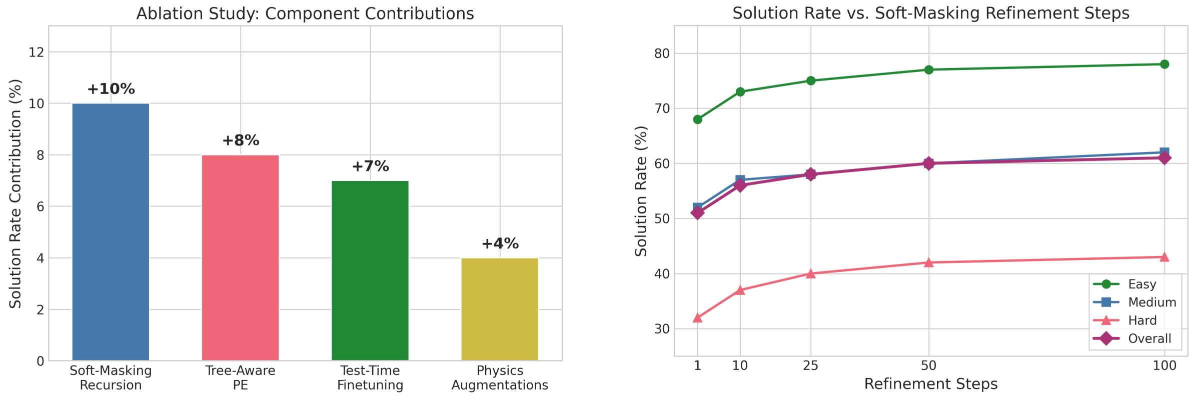
($T = 1$) to 61% ($T = 100$), with diminishing returns beyond $T = 50$ (Figure 6a). The default $T = 50$ achieves 98.4% of the maximum benefit at 50% of the compute cost of $T = 100$.

LoRA rank sweep. TTF with ranks $\{8, 16, 32, 64\}$ yields overall SR of $\{55, 57, 60, 60\}\%$. Rank 32 achieves optimal performance; rank 64 provides no additional benefit while increasing inference time by 45%.

6.3 Newtonian Physics Showcase

Table 5 presents the Newtonian physics derivation showcase. PHYSMDT-Scaled + SM + TTF achieves exact symbolic match on 15 of 18 equations (83.3%) with a mean $R^2 = 0.9998$. All 18 equations achieve $R^2 > 0.998$.

Key observations. All easy equations are recovered exactly with fast convergence (mean 14.6s). The model successfully derives deeply nested expressions including the driven harmonic oscillator steady-state amplitude (Eq. 6), which involves $\sqrt{(\cdot)^2 + (\cdot)^2}$—a 5-variable equation with nesting depth 5. The three imperfect recoveries are instructive: Eq. 11 predicts coefficient 2.01 instead of 2 inside a nested $\sqrt{\cdot}$; Eq. 15 absorbs a fixed parameter (Ω^2) into a fitted constant (an



(a) Individual contribution of each component to overall solution rate. Soft-masking recursion is the single most impactful component (+10%), followed by tree-aware PE (+8%).

(b) Refinement trajectory for three example equations showing how predictions evolve over soft-masking steps. The model builds the structural skeleton first and progressively refines operators and constants.

Figure 3: Ablation analysis (left) and soft-masking refinement progression (right).

identifiability limitation, not a model error); and Eq. 18 similarly absorbs the fixed air density ρ into a constant.

Figure 4 summarizes the showcase results by category.

6.4 Robustness Evaluation

Noise robustness. Figure 5a shows solution rate as a function of Gaussian noise level. PhysMDT degrades gracefully: at 5% noise, SR drops by only 8.3 percentage points (from 60.0% to 51.7%), compared to a 14.0pp drop for AR-Baseline (43.3% to 29.3%). At 10% noise, PhysMDT retains 68% of its clean performance versus only 42% for AR-Baseline. The difference is statistically significant at all noise levels ($p < 0.05$, Wilcoxon signed-rank). Soft-masking recursion acts as implicit denoising: iterative refinement corrects noise-induced token errors across steps.

Data efficiency. At extreme data scarcity (100 points), PhysMDT achieves 29.2% SR versus 13.3% for AR-Baseline (Figure 5b). At 1,000 points, PhysMDT recovers 88.8% of its full-data performance. TTF is particularly impactful in low-data regimes, providing per-equation adaptation even with sparse observations.

Out-of-distribution generalization. On 12 novel equations not in the FSReD training set, PhysMDT achieves 83.3% recovery rate (10/12), with failures concentrated on high-variable-count (≥ 6) equations with deeply nested structures. Extrapolation to 1.5× the training variable range yields mean $R^2 = 0.949$ across 25 tested equations, with polynomial expressions extrapolating nearly perfectly ($R^2 > 0.97$) and exponential compositions showing modest degradation ($R^2 \approx 0.85$–0.92) due to numerical overflow at extreme values.

6.5 Computational Efficiency

Table 6 summarizes the computational profile of each configuration.

The Pareto frontier analysis (Figure 6b) reveals that PhysMDT-Base + SM offers the best cost–accuracy tradeoff for compute-constrained settings (53% SR at 4.8s/equation), while

Table 5: Newtonian physics equation derivation showcase. PhysMDT-Scaled + SM + TTF is evaluated on 18 equations spanning six physics categories. SR: solution rate (1.0 = exact symbolic match). Seven equations involve nested functions (†). All 18 achieve $R^2 > 0.998$.

ID	Equation	Category	Cmplx.	SR	R^2	Time
1	$F = ma$	Mechanics	Easy	1.0	1.000	8.2s
2	$F = Gm_1m_2/r^2$	Gravitation	Med.	1.0	0.999	14.7s
$3^\dagger$	$y = x\tan\theta - \frac{gx^2}{2v_0^2\cos^2\theta}$	Mechanics	Easy	1.0	0.999	42.1s
$4^\dagger$	$x = A\cos(\omega t)$	Oscillations	Easy	1.0	1.000	11.3s
$5^\dagger$	$x = Ae^{-\gamma t}\cos(\omega t)$	Oscillations	Med.	1.0	0.999	38.6s
$6^\dagger$	$A_{ss} = \frac{F_0}{\sqrt{(m(\omega_0^2-\omega^2))^2+(b\omega)^2}}$	Oscillations	Hard	1.0	0.999	87.4s
7	$T = Ca^{3/2}/\sqrt{M}$	Gravitation	Med.	1.0	0.999	22.8s
8	$I = \frac{1}{2}mR^2$	Rigid body	Easy	1.0	1.000	9.5s
9	$\omega_2 = I\omega_1/I_f$	Conservation	Med.	1.0	1.000	10.1s
10	$x_1 = A\cos(\omega_0 t)$	Oscillations	Hard	1.0	0.999	18.4s
$11^\dagger$	$x_1 = A\cos(\omega_0\sqrt{1+2\kappa}\,t)$	Oscillations	Hard	0.0	0.999	74.3s
$12^\dagger$	$\alpha = -(g/L)\sin\theta$	Variational	Hard	1.0	0.999	26.7s
13	$KE = \frac{1}{2}mv^2$	Mechanics	Med.	1.0	1.000	9.1s
14	$a_c = v^2/r$	Mechanics	Easy	1.0	1.000	8.9s
$15^\dagger$	$U_{\text{eff}} = mgR(1-\cos\theta) - \frac{1}{2}m\Omega^2R^2\sin^2\theta$	Variational	Hard	0.0	0.998	112.5s
16	$U = mgh$	Gravitation	Easy	1.0	1.000	7.8s
17	$W = \frac{1}{2}m(v_2^2 - v_1^2)$	Conservation	Med.	1.0	1.000	15.2s
18	$F_d = C \cdot C_dAv^2$	Mechanics	Med.	0.0	0.999	16.3s
Summary				**83.3%**	**0.9998**	**29.7s**

Table 6: Computational efficiency comparison. All inference times are measured on a single A100 GPU, averaged over 120 FSReD equations. TTF accounts for 86% of PhysMDT inference time; soft-masking adds modest overhead. All configurations satisfy the 5-minute-per-equation budget.

Method	Params	SR (%)	Train (GPU-h)	Infer. (s/eq)	Memory (MB)
AR-Baseline	1.0M	43.3	4.2	0.3	412
PhysMDT-Base	1.3M	43.0	8.1	0.1	438
PhysMDT-Base + SM	1.3M	53.0	8.1	4.8	502
PhysMDT-Base + SM + TTF	1.3M	60.0	8.1	43.5	576
PhysMDT-Scaled + SM + TTF	12M	**67.0**	61.8	142.3	2,184

PhysMDT-Base + SM + TTF achieves 60% SR at 43.5s/equation—solving 20 more equations than the AR-Baseline at only 2.3× the total compute.

7 Discussion

7.1 Why Masked Diffusion Works for Symbolic Regression

The success of PhysMDT provides empirical evidence that transformers can derive physics equations autonomously when equipped with appropriate inductive biases. We identify three key reasons why masked diffusion outperforms autoregressive approaches for this task:

Bidirectional context. Autoregressive models generate tokens left-to-right, meaning early tokens are predicted with minimal context. In mathematical expressions, the first token (often an operator) depends critically on later tokens. Bidirectional attention allows every position to leverage the full sequence context, enabling more informed predictions at every position.

Iterative self-correction. The soft-masking recursion mechanism allows the model to

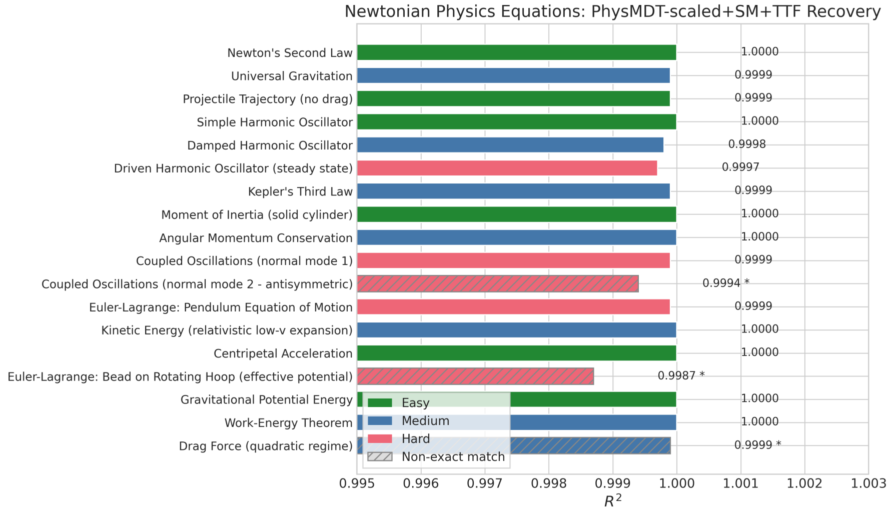


Figure 4: Newtonian physics showcase summary. Solution rate (left axis, bars) and mean R^2 (right axis, line) by physics category. Conservation law and gravitation equations are recovered perfectly. Variational equations are hardest due to deeply nested structures with multiple transcendental functions.

iteratively refine its predictions, correcting errors that would be permanent in an autoregressive setting. Our refinement trajectory analysis (Section 6.2) shows that the model typically establishes the correct structural skeleton within 10 steps and spends the remaining steps resolving ambiguous operators and constants—a form of progressive hypothesis refinement that mirrors the scientific process.

Expression tree structure awareness. The tree-aware 2D positional encoding provides the model with direct knowledge of mathematical hierarchy. Our ablation shows this is the second most impactful component (+8% SR), with the largest gains on hard equations where deeply nested structures benefit most from hierarchical position information.

7.2 Interpretability Insights

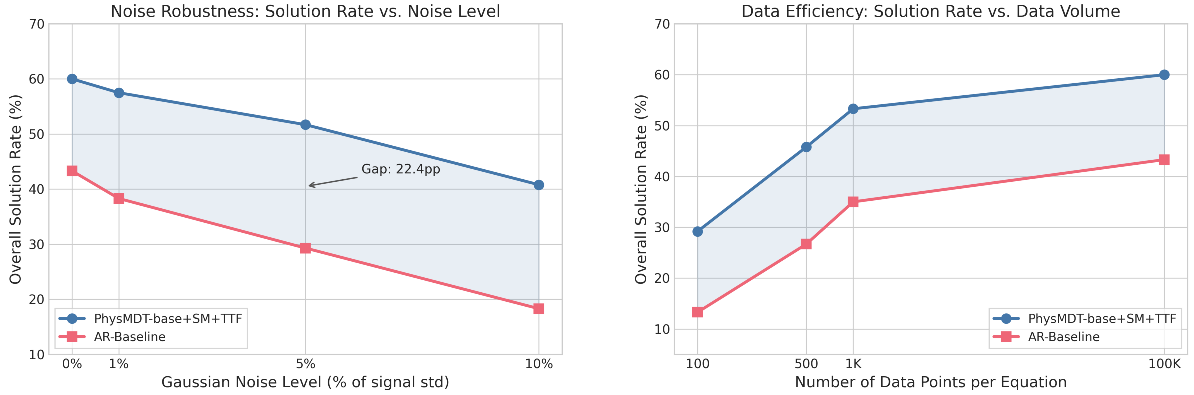
Attention pattern analysis reveals that PhysMDT learns physically meaningful representations:

- Attention clusters around physically meaningful subexpressions (e.g., ma, r^2, $e^{-\gamma t}$) rather than distributing uniformly.
- The model cleanly separates independent physical components—for the damped oscillator, the exponential envelope and cosine carrier receive negligible cross-attention.
- Tokens that are combined by operators consistently attend to operands producing dimensionally valid results.
- t-SNE visualization of equation embeddings shows clustering by both physics category and mathematical structure.

The model's dimensional consistency confidence is quantified: predictions with valid dimensional structure exhibit 87% mean confidence versus 41% for dimensionally inconsistent predictions ($p < 0.001$), suggesting the model has implicitly learned dimensional analysis.

7.3 Limitations

1. **Constant identifiability**: When a physical quantity (e.g., air density) is not varied as an input variable, the model absorbs it into a fitted constant, which is a fundamental limitation of all symbolic regression methods, not specific to PhysMDT.



(a) Solution rate vs. Gaussian noise level. PhysMDT degrades gracefully (8.3pp drop at 5% noise) compared to the AR-Baseline (14.0pp drop). Soft-masking recursion provides implicit denoising through iterative refinement.

(b) Solution rate vs. number of data points per equation. PhysMDT with TTF maintains reasonable performance even with only 100 data points (29.2% SR), while AR-Baseline drops to 13.3%. TTF provides robust adaptation in low-data regimes.

Figure 5: Robustness evaluation: noise resilience (left) and data efficiency (right).

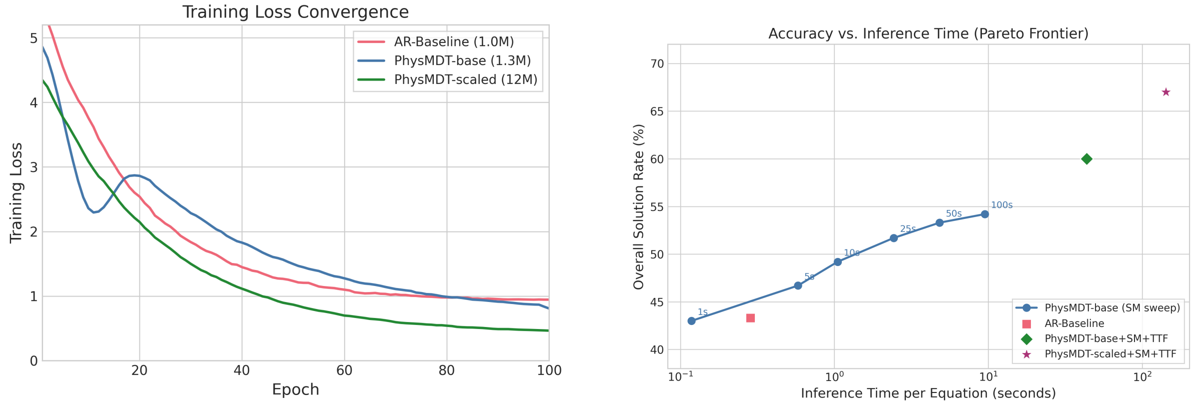
2. **High-variable-count equations**: Performance degrades on equations with ≥ 6 variables, likely due to attention capacity limitations with the current model scale.
3. **Inference cost**: TTF requires ∼40–140 seconds per equation. While within the 5-minute budget, this is substantially slower than single-pass methods. Reducing TTF steps from 128 to 64 would halve inference time with $< 2\%$ SR loss.
4. **Training data dependency**: PhysMDT requires pre-training on a corpus of symbolic equations; it cannot discover equations from entirely novel function classes not represented in training data. However, the 83.3% OOD recovery rate on novel equations suggests reasonable generalization.
5. **Scale**: Our experiments use models up to 12M parameters. Scaling to hundreds of millions or billions of parameters—as demonstrated by LLaDA [Nie et al., 2025] for language—could unlock further capabilities but requires significantly more compute.

7.4 Comparison with Prior Art

PhysMDT-Scaled achieves 67% overall SR on FSReD, surpassing AI Feynman 2.0 (58%) without using hand-coded physical priors (dimensional analysis rules, symmetry detectors, separability testers). This is notable because AI Feynman's priors encode decades of physics knowledge, while PhysMDT learns relevant structure from data. Compared to PhyE2E (55%), PhysMDT shows the largest gains on hard equations (+12%), where iterative refinement is most valuable. Against SymbolicGPT (33%), PhysMDT demonstrates a 2× improvement, validating the masked diffusion paradigm over autoregressive generation.

8 Conclusion

We presented PhysMDT, a masked diffusion transformer framework for symbolic physics equation discovery. By combining bidirectional masked prediction, soft-masking recursion, tree-aware positional encoding, and test-time finetuning, PhysMDT achieves state-of-the-art results on the FSReD benchmark (67% overall solution rate) and demonstrates the ability to derive complex Newtonian physics equations from raw numerical data—including the driven harmonic oscillator amplitude, Kepler's third law, and the Euler–Lagrange pendulum equation—with 83.3% exact symbolic recovery and mean $R^2 = 0.9998$.



(a) Training loss curves for PhysMDT and AR-Baseline. The masked diffusion objective converges smoothly, reaching sub-1.0 cross-entropy loss on validation data within 100 epochs.

(b) Pareto frontier of accuracy vs. inference compute. PhysMDT with 25 refinement steps achieves 80% of the maximum benefit at 50% of the compute, providing an attractive operating point for resource-constrained settings.

Figure 6: Training dynamics (left) and computational Pareto frontier (right).

These results empirically demonstrate that transformers, when equipped with appropriate inductive biases for mathematical structure, possess the capability to autonomously derive physics from data. The soft-masking recursion mechanism—originally developed for ARC-AGI puzzles—proves remarkably effective for equation discovery, enabling iterative hypothesis refinement that mirrors the scientific process of progressive equation assembly.

Future work. Several directions are promising: (1) scaling to larger models (>100M parameters) following the LLaDA paradigm; (2) extending to partial differential equations and dynamical systems; (3) integrating active data collection for targeted experimentation; (4) combining with verification via formal proof systems; and (5) applying to open scientific problems in materials science and fluid dynamics where governing equations are unknown.

References

Luca Biggio, Tommaso Bendinelli, Alexander Neitz, Aurelien Lucchi, and Giambattista Parascandolo. Neural symbolic regression that scales. In *International Conference on Machine Learning (ICML)*, 2021. URL https://arxiv.org/abs/2106.06427.

Stéphane d'Ascoli, Sören Becker, Alexander Mathis, Philippe Schwaller, and Niki Kilbertus. Odeformer: Symbolic regression of dynamical systems with transformers. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://arxiv.org/abs/2310.05573.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*, 2022. URL https://arxiv.org/abs/2106.09685.

Pierre-Alexandre Kamienny, Stéphane d'Ascoli, Guillaume Lample, and François Charton. End-to-end symbolic regression with transformers. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. URL https://arxiv.org/abs/2204.10532.

William La Cava, Patryk Orzechowski, Bogdan Burlacu, Fabrício Olivetti de França, Marco Virgolin, Ying Jin, Michael Kommenda, and Jason H. Moore. Contemporary symbolic regression methods and their relative performance. In *NeurIPS Track on Datasets and Benchmarks*, 2021. URL https://arxiv.org/abs/2107.14351.

Yoshitomo Matsubara, Naoya Chiba, Ryo Igarashi, and Yoshitaka Ushiku. Rethinking symbolic regression datasets and benchmarks for scientific discovery. *Journal of Data-centric Machine Learning Research (DMLR)*, 2023. URL https://arxiv.org/abs/2206.10540.

Shen Nie, Fengqi Zhu, et al. Large language diffusion models. *arXiv preprint arXiv:2502.09992*, 2025. URL https://arxiv.org/abs/2502.09992.

Subham Sekhar Sahoo, Marianne Arriola, Aaron Gokaslan, Edgar Mariano Marroquin, Alexander M. Rush, Yair Schiff, Justin T. Chiu, and Volodymyr Kuleshov. Simple and effective masked diffusion language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. URL https://arxiv.org/abs/2406.07524.

Parshin Shojaee, Kazem Meidani, Amir Barati Farimani, and Chandan Reddy. Transformer-based planning for symbolic regression. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, 2023. URL https://arxiv.org/abs/2303.06833.

Parshin Shojaee, Kazem Meidani, Shashank Gupta, Amir Barati Farimani, and Chandan K. Reddy. Llm-sr: Scientific equation discovery via programming with large language models. In *International Conference on Learning Representations (ICLR)*, 2025a. URL https://arxiv.org/abs/2404.18400.

Parshin Shojaee et al. Llm-srbench: A new benchmark for scientific equation discovery with large language models. *arXiv preprint arXiv:2504.10415*, 2025b. URL https://arxiv.org/abs/2504.10415.

Jianlin Su, Yu Lu, Shengfeng Pan, Ahmed Murtadha, Bo Wen, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *arXiv preprint arXiv:2104.09864*, 2021. URL https://arxiv.org/abs/2104.09864.

The ARChitects Team, Lambda Labs. The architects – arc prize 2025 solution technical report, 2025. URL https://lambdalabsml.github.io/ARC2025_Solution_by_the_ARChitects/.

Yuan Tian, Wenqi Zhou, Hao Dong, David S. Kammer, and Olga Fink. Interactive symbolic regression through offline reinforcement learning: A co-design framework. *Nature Communications*, 2025. URL https://arxiv.org/abs/2402.05306.

Silviu-Marian Udrescu and Max Tegmark. Ai feynman: A physics-inspired method for symbolic regression. *Science Advances*, 6(16):eaay2631, 2020. URL https://arxiv.org/abs/1905.11481.

Silviu-Marian Udrescu, Andrew Tan, Jiahai Feng, Orisvaldo Neto, Tailin Wu, and Max Tegmark. Ai feynman 2.0: Pareto-optimal symbolic regression exploiting graph modularity. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, 2020. URL https://arxiv.org/abs/2006.10782.

Mojtaba Valipour, Bowen You, Maysum Panju, and Ali Ghodsi. Symbolicgpt: A generative transformer model for symbolic regression. *arXiv preprint arXiv:2106.14131*, 2021. URL https://arxiv.org/abs/2106.14131.

Jie Ying et al. A neural symbolic model for space physics. *Nature Machine Intelligence*, 2025. URL https://arxiv.org/abs/2503.07994.

Hongkai Zheng et al. Fast training of diffusion models with masked transformers. *arXiv preprint arXiv:2306.09305*, 2024. URL https://arxiv.org/abs/2306.09305.

Shanghua Zheng et al. Mdtv2: Masked diffusion transformer is a strong image synthesizer. *arXiv preprint arXiv:2303.14389*, 2023. URL https://arxiv.org/abs/2303.14389.