EVOSLD: AUTOMATED NEURAL SCALING LAW DISCOVERY WITH LARGE LANGUAGE MODELS

Haowei Lin[1] **Xiangyu Wang**[1] **Jianzhu Ma**[2,3] **Yitao Liang**[1]
[1]Institute for Artificial Intelligence, Peking University
[2]Department of Electronic Engineering, Tsinghua University.
[3]Institute for AI Industry Research, Tsinghua University.

ABSTRACT

Scaling laws are fundamental mathematical relationships that predict how neural network performance evolves with changes in variables such as model size, dataset size, and computational resources. Traditionally, discovering these laws requires extensive human expertise and manual experimentation. We introduce EvoSLD, an automated framework for Scaling Law Discovery (SLD) that leverages evolutionary algorithms guided by Large Language Models (LLMs) to co-evolve symbolic expressions and their optimization routines. Formulated to handle scaling variables, control variables, and response metrics across diverse experimental settings, EvoSLD searches for parsimonious, universal functional forms that minimize fitting errors on grouped data subsets. Evaluated on five real-world scenarios from recent literature, EvoSLD rediscovers exact human-derived laws in two cases and surpasses them in others, achieving up to orders-of-magnitude reductions in normalized mean squared error on held-out test sets. Compared to baselines like symbolic regression and ablated variants, EvoSLD demonstrates superior accuracy, interpretability, and efficiency, highlighting its potential to accelerate AI research.[1]

1 INTRODUCTION

Scaling laws have emerged as a cornerstone of modern artificial intelligence research, providing principled guidance for designing and training large neural networks, particularly Large Language Models (LLMs) (Kaplan et al., 2020; Henighan et al., 2020). These empirical relationships describe how key performance metrics, such as loss or accuracy, scale with factors like model parameters, training data volume, and computational budget. Pioneering works, including those on compute-optimal training (Hoffmann et al., 2022) and emergent abilities (Wei et al., 2022), have demonstrated the predictive power of scaling laws, enabling researchers to forecast model behavior at unprecedented scales without exhaustive experimentation. However, deriving these laws traditionally relies on human intuition, tedious mathematical analysis, and carefully controlled experiments—a process that is time-consuming, error-prone, and often limited by domain-specific expertise.

This paper addresses the challenge of automating Scaling Law Discovery (SLD), formalizing it as an optimization problem over symbolic expressions that must generalize across varying experimental conditions. We distinguish SLD from traditional symbolic regression (Koza, 1992; Cranmer, 2023; Udrescu & Tegmark, 2020; Biggio et al., 2021; Kamienny et al., 2023) by emphasizing the separation of actively varied *scaling variables* (e.g., model size N, dataset size D) from *control variables* (e.g., architecture, optimizer) that are held constant within settings but differ across them. The objective is to uncover a single, universal closed-form expression $f(\boldsymbol{x}; \boldsymbol{\theta})$—where $\boldsymbol{x}$ are scaling variables and $\boldsymbol{\theta}$ are coefficients fitted per control group—that minimizes aggregated fitting errors while enforcing parsimony through constraints on coefficient count.

To solve this, we propose **EvoSLD**, a novel framework that reframes SLD as an evolutionary search over code subroutines. Inspired by AI-driven scientific discovery paradigms (Jumper et al., 2021; Mankowitz et al., 2023; Romera-Paredes et al., 2024), EvoSLD co-evolves two components: (1) an expression subroutine defining the symbolic structure of the law, and (2) an optimization subroutine

[1]Code is available at https://github.com/linhaowei1/SLD. This work is still in progress.

that partitions data by control variables and fits coefficients for each subset. Guided by an LLM for intelligent mutations (Kamienny et al., 2023; Rawal et al., 2024), the evolutionary loop selects, modifies, evaluates, and updates candidate programs, prioritizing those with low normalized mean squared error (NMSE) on development sets. This co-evolutionary approach not only navigates the vast hypothesis space efficiently but also adapts optimizers to handle sparse, multi-variable data—a critical advantage over fixed routines like BFGS.

Our contributions are threefold: First, we provide a rigorous formulation of SLD that aligns with scientific practice, incorporating grouped fitting, parsimony penalties, and scale-invariant evaluation metrics like NMSE and normalized mean absolute error (NMAE) on held-out test sets. Second, EvoSLD represents the first LLM-assisted system for automated scaling law discovery, demonstrating robustness across scenarios with and without explicit control variables. Third, through experiments on five diverse benchmarks from recent literature—including vocabulary size (Tao et al., 2024), supervised fine-tuning (SFT) (Lin et al., 2024; Zeng et al., 2025), domain mixtures (Ye et al., 2024), mixture-of-experts (MoE) (Krajewski et al., 2024), and data-constrained pre-training (Muennighoff et al., 2023)—we show that EvoSLD consistently outperforms baselines. It rediscovers exact human laws in vocabulary and SFT cases, surpasses them in domain mixture (NMSE: 0.0007 vs. 0.0669) and data-constrained (0.1725 vs. 0.4059) scenarios, and reveals simpler, more accurate forms in MoE, reducing NMSE by 20% while using fewer coefficients.These results underscore EvoSLD's potential to democratize and accelerate scaling law research, especially for industry practitioners. By automating the discovery process—achieving in minutes what once required weeks of human effort—EvoSLD serves as a hypothesis generator, enabling rapid iteration and verification. We also discuss limitations, such as reliance on passive data and risks of data leakage, and outline future directions toward agentic systems for active experimentation (Lehnert et al., 2024).

2 Related Work

Neural Scaling Law The performance of neural networks often follows predictable power laws relating test loss to model size and dataset size (Kaplan et al., 2020; Hoffmann et al., 2022). However, this simple picture is incomplete, with studies revealing complex phenomena like "broken" power laws, phase transitions where scaling behavior changes abruptly (Henighan et al., 2020; Michaud et al., 2023). Recent theoretical models explore these dynamics, including solvable frameworks identifying multiple phases in compute-optimal scaling (Paquette, 2024) and large-N field theory extensions beyond ridgeless limits (Zhang, 2025). This complexity has driven specialized scaling laws for scenarios such as supervised fine-tuning (Lin et al., 2024; Zeng et al., 2025), Mixture-of-Experts models (Krajewski et al., 2024), data mixtures (Ye et al., 2024), vocabulary size (Tao et al., 2024), data-constrained training (Muennighoff et al., 2023), and robotics (Sartor, 2024). We refer readers to a comprehensive survey on diverse scaling law forms (Sengupta et al., 2025).

Equation Discovery Automated equation discovery aligns with symbolic regression (SR), a core AI pursuit. Classic genetic programming (GP) evolves expressions to fit data (Koza, 1992), with modern tools like PySR advancing the field (Cranmer, 2023). AI-driven innovations include physics-inspired recursive decomposition in AI Feynman (Udrescu & Tegmark, 2020), neuro-symbolic integration (Biggio et al., 2021), and large language models for proposing equations (Kamienny et al., 2023; Wang et al., 2024). Recent developments feature interactive reinforcement learning-based co-design for large-scale SR (Tang et al., 2025), unbiased search scaling (Li et al., 2024), and LLM-based SR Shojaee et al. (2025). A survey covers these recent advances (Zhong et al., 2025).

Scientific Discovery AI AI is accelerating scientific breakthroughs through large-scale pattern recognition and intelligent search. Key successes encompass AlphaFold for protein structures (Jumper et al., 2021), AlphaDev for sorting algorithms (Mankowitz et al., 2023), FunSearch for mathematical discoveries (Romera-Paredes et al., 2024), AlphaGeometry for olympiad problems (Trinh et al., 2024), and large language models for molecular property prediction (Zheng et al., 2025). Additional frameworks address automation challenges and implications (Wang et al., 2023; Lehnert et al., 2024). Evolution-through-prompts methodologies, as in AlphaEvolve (Rawal et al., 2024), adapt LLMs to evolve solutions in scientific contexts.

3 LLM FOR SCALING LAW DISCOVERY

3.1 PROBLEM FORMULATION

Definition We formulate the problem of **Scaling Law Discovery (SLD)** as follows. Given:

- A set of n *scaling variables* $\boldsymbol{x} = (\boldsymbol{x}^{(1)}, \cdots, \boldsymbol{x}^{(n)})$, which are actively varied in an experiment (e.g., number of training tokens, model parameters).
- A set of k *control variables* $\boldsymbol{c} = (\boldsymbol{c}^{(1)}, \cdots, \boldsymbol{c}^{(k)})$, which are held constant within a given experimental setting but may differ across settings (e.g., model architecture, optimizer).
- A *response variable* $\boldsymbol{y}$ (e.g., loss, perplexity, task accuracy).
- An observed dataset $\mathcal{D} = \{(\boldsymbol{x}_i, \boldsymbol{c}_i, \boldsymbol{y}_i)\}_{i=1}^{m}$.

We partition the dataset $\mathcal{D}$ into disjoint subsets $\{\mathcal{D}_{\boldsymbol{j}}\}_{\boldsymbol{j}\in\mathcal{C}}$, where $\mathcal{C}$ is the set of unique values for the control variable $\boldsymbol{c}$, and $\mathcal{D}_{\boldsymbol{j}} = \{(\boldsymbol{x}_i, \boldsymbol{y}_i) \mid \boldsymbol{c}_i = \boldsymbol{j}\}$. The goal is to find a single, universal, closed-form expression $f(\boldsymbol{x}; \boldsymbol{\theta})$ with coefficients $\boldsymbol{\theta}$ that best describes the relationship across all control settings. The optimization problem is to find the function f^* that minimizes the total fitting error when its coefficients are tuned for each specific control condition:

$$f^* = \arg\min_{f\in\mathcal{F}} \left(\min_{\{\boldsymbol{\theta}_{\boldsymbol{j}}\}_{\boldsymbol{j}\in\mathcal{C}}} \sum_{\boldsymbol{j}\in\mathcal{C}} \mathcal{L}(f(\cdot; \boldsymbol{\theta}_{\boldsymbol{j}}); \mathcal{D}_{\boldsymbol{j}}) \right) + \lambda\Omega(f). \tag{1}$$

Here, $f(\cdot; \boldsymbol{\theta}_{\boldsymbol{j}})$ is the specific instance of the law for the control condition $\boldsymbol{j}$ with fitted coefficients $\boldsymbol{\theta}_{\boldsymbol{j}}$. The term $\Omega(f)$ penalizes the complexity of the shared symbolic structure f to ensure **parsimony**, while the loss $\mathcal{L}$ measures the **predictive fidelity** for each group. $\mathcal{F}$ is the hypothesis space of symbolic expressions.

Objective To align the SLD objective with real-world applications, we aim to minimize the fitting error, aggregated across all control groups. For each group $\mathcal{D}_{\boldsymbol{j}}$, we employ the Normalized Mean Squared Error (NMSE) as a standardized metric for the fitting error, $\mathcal{L}_{\boldsymbol{j}}$:

$$\text{NMSE}_{\boldsymbol{j}} = \frac{\sum_{i\in\mathcal{D}_{\boldsymbol{j}}} (y_i - \hat{y}_i)^2}{\sum_{i\in\mathcal{D}_{\boldsymbol{j}}} (y_i - \bar{y}_{\boldsymbol{j}})^2}, \tag{2}$$

where the sums are over points in group $\mathcal{D}_{\boldsymbol{j}}$, $\hat{y}_i = f(\boldsymbol{x}_i; \boldsymbol{\theta}_{\boldsymbol{j}})$ is the predicted value, and $\bar{y}_{\boldsymbol{j}}$ is the mean of the observed responses in that group. The total loss in Eq. (1) is the sum of these individual group losses, $\sum_{\boldsymbol{j}\in\mathcal{C}} \mathcal{L}_{\boldsymbol{j}}$.

To avoid complex tuning of the regularization parameter λ, we adopt a simple yet effective proxy for the complexity penalty $\Omega(f)$. We impose a hard constraint on the number of free coefficients within the base function f:

$$\Omega(f; \tau) = \begin{cases} 0 & \text{if the number of coefficients in } f \leq \tau \\ +\infty & \text{otherwise,} \end{cases} \tag{3}$$

where τ is a predefined threshold. This approach simplifies the optimization problem by focusing the search on functions that are inherently parsimonious.

Evaluation To provide an insightful evaluation, we assess the generalization performance of a discovered scaling law on unseen data. The full dataset $\mathcal{D}$ is partitioned into a development set, $\mathcal{D}_{\text{dev}}$, and a held-out test set, $\mathcal{D}_{\text{test}}$. This split is typically stratified by the control variable $\boldsymbol{c}$ to ensure that different conditions are represented in both sets.

The discovery process is performed on $\mathcal{D}_{\text{dev}}$:

1. A candidate law f is proposed.
2. For each control group $\boldsymbol{j}$ present in $\mathcal{D}_{\text{dev}}$, the optimal coefficients $\boldsymbol{\theta}^*_{\boldsymbol{j}}$ are found by fitting $f(\boldsymbol{x}; \boldsymbol{\theta}_{\boldsymbol{j}})$ to the data subset $\mathcal{D}_{\text{dev},\boldsymbol{j}}$.

3. The overall quality of f is judged by the total loss (e.g., sum of $\text{NMSE}_{\boldsymbol{j}}$) across all groups in $\mathcal{D}_{\text{dev}}$. This process is repeated to find the optimal law f^*.

For the final evaluation on $\mathcal{D}_{\text{test}}$, we use the discovered law f^* and the coefficients $\{\boldsymbol{\theta}_{\boldsymbol{j}}^*\}$ learned from the development set. For each test point $(\boldsymbol{x}_i, \boldsymbol{c}_i, \boldsymbol{y}_i) \in \mathcal{D}_{\text{test}}$, the prediction is made using the coefficients corresponding to its control group: $\hat{\boldsymbol{y}}_i = f^*(\boldsymbol{x}_i; \boldsymbol{\theta}_{\boldsymbol{c}_i}^*)$. We then compute normalized regression metrics over the entire test set to provide a scale-invariant assessment of performance. The normalization is performed with respect to the statistics of the combined development and test sets, $\mathcal{D}_{\text{total}} = \mathcal{D}_{\text{dev}} \cup \mathcal{D}_{\text{test}}$.

- **Normalized Mean Squared Error (NMSE):** This metric normalizes the Mean Squared Error on the test set by the variance of the response variable over the entire dataset. A lower NMSE indicates a better fit, with a value of 1.0 corresponding to a model that performs no better than predicting the mean.

$$\text{NMSE} = \frac{\sum_{i \in \mathcal{D}_{\text{test}}} (y_i - \hat{y}_i)^2 / |\mathcal{D}_{\text{test}}|}{\sum_{i \in \mathcal{D}_{\text{total}}} (y_i - \bar{y}_{\text{total}})^2 / |\mathcal{D}_{\text{total}}|} \tag{4}$$

  where $\bar{y}_{\text{total}}$ is the mean of the response variable y across all data points in $\mathcal{D}_{\text{total}}$.

- **Normalized Mean Absolute Error (NMAE):** Similarly, this metric normalizes the Mean Absolute Error on the test set by the mean absolute deviation of the response variable over the entire dataset. It is less sensitive to outliers than NMSE.

$$\text{NMAE} = \frac{\sum_{i \in \mathcal{D}_{\text{test}}} |y_i - \hat{y}_i| / |\mathcal{D}_{\text{test}}|}{\sum_{i \in \mathcal{D}_{\text{total}}} |y_i - \bar{y}_{\text{total}}| / |\mathcal{D}_{\text{total}}|} \tag{5}$$

These normalized metrics are preferred for final evaluation as they provide a consistent measure of goodness-of-fit that is independent of the scale of the response variable $\boldsymbol{y}$, facilitating comparison across different problems.

3.2 Comparison with Traditional Symbolic Regression

Our SLD formulation represents a targeted approach that differs significantly from traditional symbolic regression (SR). A conventional SR method would typically treat all variables—both our scaling variables $\boldsymbol{x}$ and control variables $\boldsymbol{c}$—as inputs to a single function. The goal would be to discover one comprehensive equation $f(\boldsymbol{x}, \boldsymbol{c})$ that directly models the response variable across the entire dataset.

In contrast, our approach is designed to uncover a more fundamental scientific principle. It searches for a universal symbolic form, $f(\boldsymbol{x}; \boldsymbol{\theta})$, that remains constant across different experimental conditions. The influence of the control variables is captured by fitting the coefficients $\boldsymbol{\theta}$ independently for each data subset defined by a unique control setting $\boldsymbol{j}$. This methodology, formalized in Equation 1, more closely mirrors scientific practice, where the aim is to find a general law whose specific parameters are modulated by the experimental context.

This distinction is critical. The traditional SR approach of finding a single, all-encompassing function $f(\boldsymbol{x}, \boldsymbol{c})$ can introduce unnecessary complexity. The control variable $\boldsymbol{c}$ might be woven into the final expression in an intricate, non-intuitive way. Discovering such a complex, high-dimensional function reliably would likely require a very large and comprehensive dataset to avoid finding spurious relationships. More importantly, this complexity is often not what scientists are looking for. The underlying physical law is often simple, and our SLD formulation is tailored to find this parsimonious and generalizable structure by separating the core phenomenon from contextual parameterizations.

3.3 Law Discovery via Code Optimization

Inspired by recent successes in AI-driven scientific discovery, which leverage evolutionary search and represent scientific artifacts as code, we introduce **EvoSLD**, a framework for automated scaling law discovery. EvoSLD reframes the search for a scaling law f^* as an optimization problem over programs, guided by an evolutionary algorithm. The overall process is illustrated in Figure 1.

The core idea is to decompose the problem from Sec. 3.1 into two co-evolving components, which are represented as code subroutines:

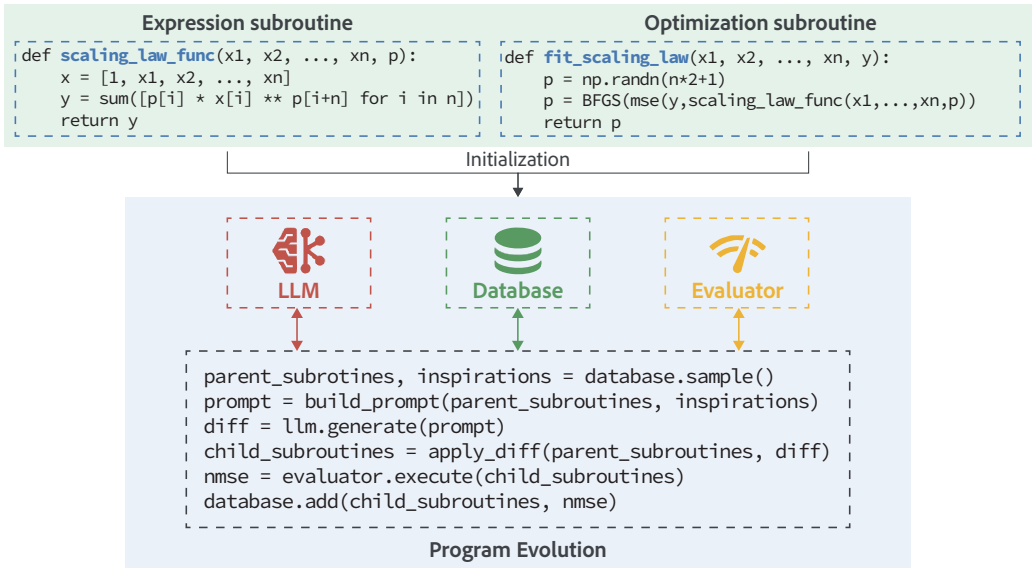

Figure 1: An overview of the **EvoSLD** framework. The evolutionary algorithm maintains a database of candidate programs, each comprising an expression and an optimizer subroutine. In each iteration, a parent candidate is selected from the database. An LLM proposes a mutation to either the expression or the optimizer code, creating a child candidate. The child is then evaluated on the dataset $\mathcal{D}$, and the resulting (program, score) pair is added to the database. This loop continues until a termination condition is met.

1. **The Expression Subroutine:** This defines the symbolic structure of the scaling law, $f(\boldsymbol{x}; \boldsymbol{\theta})$, with undefined coefficients $\boldsymbol{\theta}$. This structure represents a family of potential laws.
2. **The Optimization Subroutine:** This routine takes the expression subroutine and the dataset $\mathcal{D}$ as input. It partitions the data into groups $\{\mathcal{D}_{\boldsymbol{j}}\}$ based on the control variables $\boldsymbol{c}$. For each group $\boldsymbol{j}$, it then finds the optimal coefficients $\boldsymbol{\theta}^*_{\boldsymbol{j}}$ that best fit the expression to that specific data subset, thereby solving the inner minimization problem of Eq. (1).

The overall discovery process is an evolutionary search that optimizes these two subroutines concurrently to find the globally optimal law f^*. As shown in fig. 1, the algorithm proceeds as follows:

Initialization The process begins with a population of initial candidates stored in a database. Each candidate is a pair of subroutines: a simple expression (e.g., a power law $f(\boldsymbol{x}; \boldsymbol{\theta}) = \sum_i \theta_i \boldsymbol{x}_i^{\theta_{n+i}} + \theta_0$) and a standard optimization algorithm (e.g., BFGS). Each candidate is evaluated by executing its optimizer to fit the expression to each data subset $\mathcal{D}_{\boldsymbol{j}}$ and aggregating the group-wise errors to get a total objective score.

Evolutionary Loop The algorithm then iterates through the mutation and selection loop under an "AlphaEvolve-style" framework:

(a) **Selection:** A "parent" pair of subroutines is sampled from the database, prioritizing those with better objective scores.

(b) **Mutation with LLM:** The selected parent pair, along with other high-performing pairs from the database, are used to construct a prompt for an LLM. The prompt instructs the LLM to suggest modifications to the parent's code (either the expression or the optimizer) to create a "child" pair that is likely to yield better performance.

(c) **Evaluation:** The new child's subroutines are executed. The optimization subroutine fits the expression to each control group $\mathcal{D}_{\boldsymbol{j}}$ in the development set, yielding a set of optimal coefficients $\{\boldsymbol{\theta}^*_{\boldsymbol{j}}\}$. The performance of the law is then measured by the total fitting error (e.g., sum of NMSE scores) across all groups.

(d) **Update:** The new child pair and its score are added to the database.

This loop continues until a predefined termination condition is met (e.g., reached a predefined number of iterations).

Table 1: An overview of the five scaling law discovery scenarios. For each scenario, we list the control variables that define distinct experimental groups, the number of free coefficients (τ) allowed in the symbolic expression as per Eq. (3), and the naive power law used to initialize the evolutionary search. The value of τ is set to match the complexity of the corresponding human-designed law.

Scenario	# Allowed Coefficients (τ)	Control Variables (c)	Initialized Law (Naive Power Law)
Vocabulary Size	7	None	$L(N,V,D) = k_1N^{\alpha_1} + k_2V^{\alpha_2} + k_3D^{\alpha_3} + C$ L: unigram normalized loss; N: non-vocab params; V: vocab size; D: dataset size
Supervised Fine-tuning	4	Model Architecture, Dataset Task	$L(D) = kD^{\alpha} + A$ L: fine-tuning loss; D: training data size
Domain Mixture	$M(M+2)$	Model Size	$L_i(\mathbf{r}) = A_i + B_i \sum_{j=1}^{M} r_j^{\alpha_j}$ L_i: loss for domain i; $\mathbf{r} = (r_1, \ldots, r_M)$: data mixture proportions.
Mixture-of-Experts	6	None	$L(N,E) = k_1N^{\alpha_1} + k_2E^{\alpha_2} + C$ L: loss; N: number of dense parameters; E: number of experts.
Data-Constrained Pre-training	7	None	$L(N,D,U) = C + A_1N^{\alpha_1} + A_2D^{\alpha_2} + A_3U^{\alpha_3}$ L: loss; N: model params; D: training tokens; U: unique tokens.

Termination Once the process terminates, the expression f from the highest-scoring pair in the database is returned as the final discovered scaling law, f^*.

To better simulate the process of scientific discovery by humans, during the prompt construction phase, we provide the LLM with the background description of the scaling law study and the statistics of the collected data (e.g., the number of data points, the range of each variable).

4 Experiments

To validate the effectiveness of our proposed **EvoSLD** framework, we conduct experiments across five distinct scaling law discovery scenarios from recent literature. Following our problem formulation (Sec. 3.1), our primary goal is to assess whether EvoSLD can discover a single, universal symbolic expression that, after fitting coefficients for each control setting, accurately describes the underlying scaling behavior. We aim to rediscover or even improve upon the original, human-derived scaling laws and compare EvoSLD's performance against several baseline methods.

4.1 Experimental Setup

Datasets and Scenarios. We evaluate EvoSLD on five scenarios from recent literature, each presenting unique challenges with different scaling and control variables (Table 1). For each scenario, we compare the discovered laws against the original, human-designed expressions, which are listed below:

1. **Vocabulary Scaling Law** (Tao et al., 2024):
$$L(N,V,D) = \frac{A}{N^{\alpha}} + \frac{B}{V^{\beta}} + \frac{C}{D^{\gamma}} + E$$
2. **SFT Scaling Law** (Lin et al., 2024; Zeng et al., 2025):
$$L(D) = \frac{A}{D^{\alpha} + B} + C$$
3. **Domain Mixture Scaling Law** (Ye et al., 2024):
$$L_i(\mathbf{r}) = c_i + k_i \exp\left(\sum_{j=1}^{M} t_{ij} r_j\right)$$
4. **Mixture-of-Experts (MoE) Scaling Law** (Krajewski et al., 2024):
$\log L(N,E) = a \log N + b \log \hat{E} + c \log N \log \hat{E} + d$, where $\frac{1}{\hat{E}} = \frac{1}{E-1+\left(\frac{1}{E_{\text{start}}} - \frac{1}{E_{\text{max}}}\right)^{-1}} + \frac{1}{E_{\text{max}}}$
5. **Data-Constrained Scaling Law** (Muennighoff et al., 2023):
$$L(N,D,U) = E + \frac{A}{\left(U_N + U_N R_N^* \left(1 - \exp\left(-\frac{\frac{N}{U_N} - 1}{R_N^*}\right)\right)\right)^{\alpha}} + \frac{B}{\left(U + U R_D^* \left(1 - \exp\left(-\frac{\frac{D}{U} - 1}{R_D^*}\right)\right)\right)^{\beta}}$$

Evolutionary Search Configuration. Each evolutionary search is initialized with a naive power law as specified in Table 1 and a BFGS coefficient optimization subroutine. For the evolutionary loop, we use the OpenEvolve framework[2], with `o4-mini-2025-04-16` serving as the base LLM for the mutation step. The search runs for 50 generations using a multi-population setup of 3 separate islands, between which a 10% migration occurs every 20 generations.

Development and Test Set Split. Data for all scenarios is sourced from the official releases of the original papers. Our evaluation protocol adheres to the formulation in Sec. 3.1, with data splits designed to test generalization. For scenarios with **control variables** (**SFT** and **Domain Mixture**), the data is partitioned based on these variables. For the **SFT** scenario, where "(Model Architecture, Dataset Task)" is the control variable, we have 42 distinct control groups. For each group, we use data points with training sizes up to 409,600 for the development set ($\mathcal{D}_{\text{dev}}$) and reserve the largest size (819,200) for the test set ($\mathcal{D}_{\text{test}}$). For the **Domain Mixture** scenario, with "Model Size" as the control variable, we use four model sizes (70M, 160M, 305M, 410M) as four control groups. For each group, 20 mixture proportions form the development set, and 6 are held out for the test set. For the remaining three scenarios that lack explicit control variables, we treat the entire dataset as a single group and perform a standard 80/20 random split for the development and test sets, respectively, using a fixed random seed (42) for reproducibility.

SLD Baselines. We benchmark EvoSLD against two classes of baselines. The first class follows our problem formulation (Sec. 3.1), where a universal symbolic form is found and its coefficients are fitted for each control group. The second class comprises traditional symbolic regression (SR) methods, which treat all variables (scaling and control) as inputs to a single monolithic function, as discussed in Sec. 3.2.

- **Naive SLD**: This baseline uses the simple power law specified for each scenario in Table 1. It follows our proposed methodology by fitting the coefficients of this fixed expression for each control group. Its performance establishes a lower bound, isolating the benefit of the evolutionary discovery process.
- **PySR** (Cranmer, 2023): A high-performance symbolic regression library that utilizes regularized evolution. As a traditional SR method, it seeks a single monolithic function. We configure it with 20 iterations, 31 populations of size 27, and an L1 loss for fitting. The operator set includes binary operators $\{+, -, \times, /, \min, \max\}$ and unary operators $\{\sqrt{\cdot}, \log, \text{abs}, \text{neg}, \text{inv}\}$.
- **GPlearn** (Koza, 1992): A Genetic Programming (GP) symbolic regression method. Like PySR, it is treated as a traditional SR method. We used the `SymbolicRegressor` with a population of 1000 evolved for 20 generations, a parsimony coefficient of 0.001, and a standard operator set $\{+, -, \times, \div, \sqrt{\cdot}, \log, \exp, \text{abs}, \max, \min, \text{inv}\}$.
- **EvoSLD (w/o opt.)**: This is an ablation of our method to highlight the benefit of co-evolving the optimizer. It is also an extension of LLMSR, an LLM-based SR technique for SLD task. It uses the same EvoSLD framework to evolve the symbolic expression with an LLM. However, the optimization subroutine is fixed to a standard BFGS optimizer with `np.ones` initialization for coefficients following LLMSR, rather than being co-evolved. It employs the same grouped fitting strategy as the full EvoSLD.
- **Human**: This baseline uses the original scaling law expression from the source paper. We fit its coefficients using the same grouped optimization procedure as EvoSLD. This allows for a fair comparison of the symbolic forms themselves.[3]

4.2 Experimental Results

Our experimental results, summarized in Table 2, demonstrate the effectiveness of EvoSLD in autonomously discovering scaling laws across diverse and complex scenarios. The analysis below delves into key comparisons that highlight the strengths of our approach.

[2] https://github.com/codelion/openevolve

[3] Scaling laws are typically fitted by humans using data from controlled experiments. Our passive data collection, however, necessitates direct multivariable optimization—a considerably more challenging task. For a fair comparison, we apply a sophisticated optimization method to the human baseline rather than a naive one.

Table 2: Main experimental results on the test sets for all five scenarios. We report NMSE and NMAE, as defined in Sec. 3.1. The best and second best performance for each metric in each scenario are highlighted in **bold** and underlined, respectively. Results are averaged over three runs. "NA" denotes the method is not suitable for SLD with control variables; "-" means EvoSLD discovered law is exactly the same expression as the human designed.

	EvoSLD (Ours)		**Naive SLD**		**PySR**		**GPlearn**		**EvoSLD (w/o opt.)**		**Human**	
Scenario	NMSE	NMAE	NMSE	NMAE	NMSE	NMAE	NMSE	NMAE	NMSE	NMAE	NMSE	NMAE
Vocabulary Size	**0.0242**	**0.1479**	1.0003	1.0189	32.2138	7.4678	32.2138	7.4678	0.0254	0.1642	-	-
Supervised Fine-tuning	**0.0058**	**0.0648**	1.4484	1.3374	NA	NA	NA	NA	0.0484	0.1306	-	-
Domain Mixture	**0.0007**	**0.0228**	1.1852	0.9851	NA	NA	NA	NA	0.0023	0.0507	0.0669	0.2557
Mixture-of-Experts	**0.0167**	**0.1243**	23.3949	5.6000	0.0270	0.1602	0.8662	0.8644	0.2040	0.1290	0.0209	0.1370
Data-Constrained	**0.1725**	**0.3098**	0.7763	0.8919	1.1573	1.1102	0.4661	0.5274	0.7763	0.8919	0.4059	0.6139

Rediscovering and Surpassing Human-Designed Laws. A remarkable finding is EvoSLD's ability to not only match but also surpass the performance of human-designed laws. For the Vocabulary Size and SFT scenarios, EvoSLD rediscovers the *exact* symbolic expressions derived by human experts, indicated by the "-" in the "Human" baseline column in Table 2. Furthermore, in more complex scenarios, EvoSLD discovers laws that provide a *superior fit to the data* compared to their human-designed counterparts. For instance, in the Domain Mixture and Data-Constrained scenarios, EvoSLD reduces the NMSE by orders of magnitude (from 0.0669 to 0.0007 and 0.4059 to 0.1725, respectively). This suggests that for intricate, multi-variable systems, an evolutionary search guided by an LLM can explore a wider and more promising space of functional forms than human intuition alone. It's worth noting that this advantage is observed on passively collected data; we will discuss the implications of this data collection method in Sec. 5 (limitations).

The Failure of Traditional Symbolic Regression. Traditional SR methods, such as PySR and GPlearn, perform poorly across all tested scenarios. As shown in Table 2, they either fail to find any meaningful relationship, resulting in extremely high error (e.g., NMSE of 32.2 for PySR on Vocabulary Size), or are simply not applicable (NA) to scenarios with control variables. This failure stems from their lack of domain prior knowledge. Given a set of operators (e.g., $10+$ operators in our baselines) and input variables, the search space of possible equations grows combinatorially, making an unguided search intractable. Without strong priors on the structure of the scaling law, these methods tend to discover overly complex, uninterpretable expressions that generalize poorly. In contrast, EvoSLD leverages the rich priors embedded in LLMs to navigate this vast search space effectively, focusing on plausible and well-structured candidate laws.

The Critical Role of Co-evolution. The comparison between the full EvoSLD and the EvoSLD (w/o opt.) ablation starkly illustrates the benefits of our co-evolutionary framework. In every scenario, co-evolving the optimization subroutine alongside the symbolic form yields a significant performance improvement. This is particularly evident in the SFT and MoE scenarios, where the NMSE is reduced by nearly an order of magnitude. This highlights a crucial challenge in scaling law discovery: *optimizing the coefficients* of a complex, multi-variable function from sparse data is a demanding task. A fixed, general-purpose optimizer like BFGS often falls short. By co-evolving a specialized optimization strategy, EvoSLD finds significantly better parameter fits, leading to more accurate laws.

The accuracy gained from superior optimization is not merely academic; it has critical downstream applications. For example, the precise coefficients of an SFT scaling law are essential for effective model selection (Lin et al., 2024), while the parameters of a pre-training scaling law are used to determine compute-optimal pre-training configurations (Hoffmann et al., 2022). Therefore, the enhanced fitting provided by co-evolution is vital for making scaling laws practically useful.

4.3 Case Study: MoE Scaling Law

Figure 2 compares the laws recovered by each method for the challenging Mixture-of-Experts (MoE) setting. We mainly focus on the effect that number of experts (E) play on test loss (L), by fixing dense model parameters (N).

PySR's Artefactual Law. PySR converges to a piecewise expression involving `max` operators to capture the plateau effect when E is small. This results in an unphysical *sharp kink* in the loss curve, violating the smooth power-law behaviour expected in scaling studies. While its numerical error is

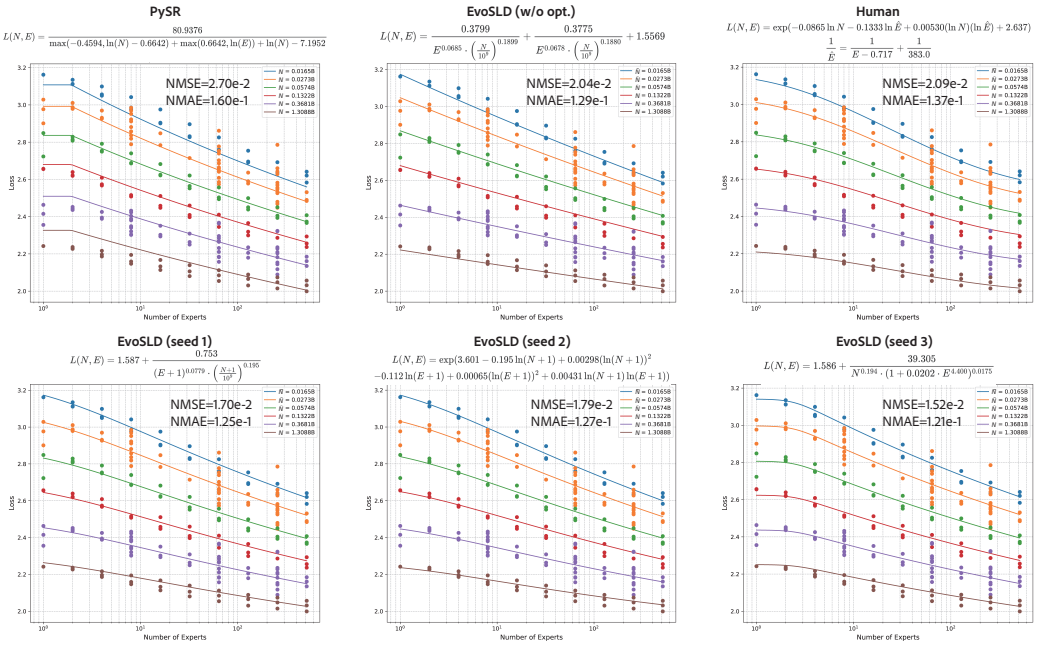

Figure 2: **Comparison of MoE scaling laws.** Each panel plots the observed loss (dots) of a Transformer-MoE model against the number of experts E for a fixed number of dense parameters N. Solid lines show predictions from the symbolic law displayed above each panel. Insets report NMSE/NMAE on the test set. *Top row:* PySR (left) finds an unintuitive law with poor fit. EvoSLD without a co-evolving optimizer (middle) and the human-designed law (right) perform better but are surpassed by the full EvoSLD. *Bottom row:* Three independent EvoSLD runs discover simpler, more accurate laws. Seed 1 (left) achieves the best error with only four of seven allowed free coefficients.

not catastrophic (NMSE $= 2.7 \times 10^{-2}$), this outcome highlights the danger of unguided symbolic regression in a large operator space.

Human Law Verification. As shown in Table 3, our fitted parameters closely match the originally reported values fron the MoE scaling law paper, confirming the correctness of our fitting routine and the fairness of the "Human" baseline.

Table 3: Fitted coefficients for human-designed MoE law. Our fit is compared to the values from the paper (Krajewski et al., 2024).

Setting	a	b	c	d	E_{start}	E_{max}
S-BASE (paper)	-0.082	-0.108	0.009	1.104	1.85	314
RL-R (paper)	-0.083	-0.126	0.012	1.111	1.88	470
HASH (paper)	-0.087	-0.136	0.012	1.157	4.18	478
Our fit	-0.087	-0.133	0.005	2.637	3.50	383

Physical Interpretation and Bounding Effects. A key challenge is modeling the diminishing returns from adding more experts. Both the human-designed law and EvoSLD's discoveries successfully capture this **bounding effect**, but they do so through distinct, physically meaningful mechanisms that imply different asymptotic behaviors.

- **Human Law: Bounding an Input Variable.** The human law models saturation by transforming the raw number of experts E into an *effective* count $\hat{E}$. This is achieved via a harmonic mapping that ensures $\hat{E}$ smoothly approaches a maximum value, E_{max}, as E becomes very large:
$$\frac{1}{\hat{E}} \approx \frac{1}{E} + \frac{1}{E_{\text{max}}}$$
As a result, the loss $L(N, E)$ does not decrease indefinitely as more experts are added. Instead, for a fixed model size N, the loss approaches an **N-dependent lower bound**. This bound represents the best possible performance for a given dense parameter count N, according to this law.
- **EvoSLD: Introducing a Global Irreducible Loss.** In contrast, EvoSLD discovers a different mechanism common in scaling literature. Two of the three runs found laws

incorporating a **constant, irreducible loss term**, L_∞:

$$L(N, E) = L_\infty + f(N, E)$$

Here, $f(N, E)$ is a term that decays as either the model size N or the number of experts E grows. This formulation implies that the loss approaches the same **constant floor** L_∞ regardless of how large N and E become. This L_∞ represents a fundamental, task-dependent limit on performance (i.e., the Bayes error), a common assumption in scaling studies.

Interestingly, the law from EvoSLD-2, an exponential of a bi-quadratic polynomial, lacks an explicit saturation term. While it fits the training data well, it might extrapolate poorly for large E, underscoring the importance of encoding correct physical priors.

Compactness of Discovered Laws. Despite a budget of seven free coefficients, the best-performing law (EvoSLD-1) uses only *four*, yet it outperforms all baselines (NMSE $= 1.52\times10^{-2}$ vs. 2.09×10^{-2} for the human law). This parsimony stems from our LLM-guided mutation operators, which favour simplicity. The result supports our central claim: *co-evolving symbolic forms and their optimizers yields laws that are more accurate, interpretable, and robust.*

5 Discussion and Limitations

On the Possibility of Data Leakage. A valid concern is whether the scaling laws discovered by EvoSLD are simply retrieved from the LLM's pre-training data, which likely contains scientific papers discussing such laws. However, several pieces of evidence suggest this is not the case. Most notably, for a single problem like the MoE scaling law, independent runs of EvoSLD discover diverse yet highly effective symbolic forms (see Figure 2). If the process were simple memorization, we would expect convergence to a single, canonical formula from the literature. Instead, the diversity of solutions indicates that the LLM is engaging in genuine problem-solving, leveraging its understanding of mathematical and physical principles to construct novel candidates within the evolutionary framework.

Challenges of Passive Data Collection. Our current methodology operates on a passive data collection pipeline, using datasets sourced directly from published papers. This limits our ability to perform new, controlled experiments to gather additional data points, which is a key part of the traditional scientific process. Without the freedom to design experiments—for instance, by holding certain scaling variables constant to simplify the law and isolate coefficients—the task of fitting a multi-variable function becomes significantly more challenging. This places a heavier burden on the optimization subroutine. Nevertheless, EvoSLD's success in recovering the MoE law, which was originally fitted using data from carefully controlled experiments, demonstrates that our co-evolutionary optimization is robust enough to overcome this limitation.

Future Work: Towards Agent-Based Discovery. The current EvoSLD framework confines the discovery process to a relatively static pipeline: an evolutionary search that mutates two predefined subroutines. A natural and powerful extension would be to develop a more open-ended, agent-based system for scientific discovery. Such an agent could be equipped with a broader set of actions, such as writing and executing arbitrary code in a sandboxed environment, searching online for relevant theories or data, and performing sophisticated data analysis through visualization or statistical testing. This more autonomous approach would not only better simulate the multifaceted nature of human scientific inquiry but could also directly address the limitation of passive data collection by enabling the agent to actively propose which new data points would be most informative to collect.

Accelerating Scientific Discovery. Our results show that EvoSLD has significant potential to accelerate the pace of scientific discovery. For instance, it consistently rediscovers the SFT scaling law in the exact form proposed by Lin et al. (2024). While the original discovery required "tedious human brainstorming and... careful mathematical analysis," (acknowledged by the authors) EvoSLD accomplishes the same feat in under a minute. This dramatic speed-up highlights the framework's power as a tool for hypothesis generation and validation. We believe EvoSLD can become an invaluable assistant for researchers and engineers, particularly for LLM developers in industry, allowing them to rapidly discover and verify scaling behaviors for their proprietary models and datasets, thereby significantly accelerating their development and optimization cycles.

REFERENCES

Luca Biggio, William Lacava, Fabien Laguillaumie, Forrest N Rueden, Tim Rocktäschel, and Yarin Gal. Neural-symbolic regression that scales. In *International Conference on Machine Learning*, pp. 876–886. PMLR, 2021. 1, 2

Miles Cranmer. PySR: A high-performance and parallelized symbolic regression package in python/julia. *arXiv preprint arXiv:2305.01582*, 2023. 1, 2, 7

Tom Henighan, Jared Kaplan, Mor Katz, and Mark Chen. Scaling laws for autoregressive generative modeling. In *arXiv preprint arXiv:2010.14701*, 2020. 1, 2

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Jardar Uesato, et al. Training compute-optimal large language models. In *Advances in Neural Information Processing Systems*, volume 35, pp. 30448–30462, 2022. 1, 2, 8

John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, et al. Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589, 2021. 1, 2

Pierre-Alexandre Kamienny, Guillaume Lample, and François Charton. Symbolic regression with large language models. *arXiv preprint arXiv:2306.10256*, 2023. 1, 2

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Chris Hallacy, Jan Leike, Aditya Ramesh, et al. Scaling laws for neural language models. In *arXiv preprint arXiv:2001.08361*, 2020. 1, 2

John R Koza. *Genetic programming: on the programming of computers by means of natural selection*, volume 1. MIT press, 1992. 1, 2, 7

Jakub Krajewski, Jan Ludziejewski, Kamil Adamczewski, Maciej Pióro, Michał Krutul, Szymon Antoniak, Kamil Ciebiera, Krystian Król, Tomasz Odrzygóźdź, Piotr Sankowski, et al. Scaling laws for fine-grained mixture of experts. *arXiv preprint arXiv:2402.07871*, 2024. 2, 6, 9

Robert Lehnert, Bernhard Voigtl"ander, and Kai Sundmacher. Automating the practice of science: Opportunities, challenges, and implications. *Proceedings of the National Academy of Sciences*, 121(31):e2401238121, 2024. doi: 10.1073/pnas.2401238121. 2

Jordan Ran Li, Joao F. D. S. Martins, Pedro Carvalho, Ricardo Ferreira, Hugo Lemos, and Jacek Gondzio. Scaling up unbiased search-based symbolic regression. pp. 4284–4292, 2024. 2

Haowei Lin, Baizhou Huang, Haotian Ye, Qinyu Chen, Zihao Wang, Sujian Li, Jianzhu Ma, Xiaojun Wan, James Zou, and Yitao Liang. Selecting large language model to fine-tune via rectified scaling law. *arXiv preprint arXiv:2402.02314*, 2024. 2, 6, 8, 10

Daniel J Mankowitz, Andrea Michi, Anton Zhernov, Marco Gelmi, Marco Selvi, Cosmin Paduraru, Edouard Leurent, Máté Ganea, George Toderici, Grzegorz Kopp, et al. Faster sorting algorithms discovered using deep reinforcement learning. *Nature*, 620(7972):83–88, 2023. 1, 2

Eric J Michaud, Ziming Liu, and Max Tegmark. The quantization model of neural scaling. *arXiv preprint arXiv:2309.15503*, 2023. 2

Niklas Muennighoff, Alexander Rush, Boaz Barak, Teven Le Scao, Nouamane Tazi, Aleksandra Piktus, Sampo Pyysalo, Thomas Wolf, and Colin A Raffel. Scaling data-constrained language models. In *Advances in Neural Information Processing Systems*, volume 36, pp. 50358–50376, 2023. 2, 6

Courtney Paquette. 4+3 phases of compute-optimal neural scaling laws, 2024. 2

Aayush Rawal, Arjun Kumar, Sankalan Parthasarathy, Khaled S Refaat, Sergey Levine, and Chelsea Finn. Language models can generate code for evolving programs. *arXiv preprint arXiv:2402.15437*, 2024. 2

Bernardino Romera-Paredes, Mohammadamin Barekatain, Alexander Novikov, Matej Balestriero, Chris M Kumar, Emilien Dupont, Francisco J Ruiz, Jordan Ellenberg, Pengming Wang, Omar Fawzi, et al. Mathematical discoveries from program search with large language models. *Nature*, 625(7995):468–475, 2024. 1, 2

Sebastian Sartor. Neural scaling laws in robotics, 2024. 2

Ayan Sengupta, Yash Goel, and Tanmoy Chakraborty. How to upscale neural networks with scaling law? a survey and practical guidelines. *ArXiv*, abs/2502.12051, 2025. URL https://api.semanticscholar.org/CorpusID:276421468. 2

Parshin Shojaee, Ngoc-Hieu Nguyen, Kazem Meidani, Amir Barati Farimani, Khoa D Doan, and Chandan K Reddy. Llm-srbench: A new benchmark for scientific equation discovery with large language models. *arXiv preprint arXiv:2504.10415*, 2025. 2

Lu Tang, Bingdong Zhang, Li Lin, Zhenyu Pan, Zhichao Luo, Chengyang Peng, Sherryl Zheng, and Kay Chen Tan. Symbolic q-network: A co-design approach for large-scale symbolic regression. *Nature Communications*, 2025. doi: 10.1038/s41467-025-59288-y. 2

Chaofan Tao, Qian Liu, Longxu Dou, Niklas Muennighoff, Zhongwei Wan, Ping Luo, Min Lin, and Ngai Wong. Scaling laws with vocabulary: Larger models deserve larger vocabularies. *Advances in Neural Information Processing Systems*, 37:114147–114179, 2024. 2, 6

Trieu H Trinh, Yuhuai Wu, Quoc V Le, Minh-Thang Luong, Thang Ma, and Mohammad Norouzi. Solving olympiad geometry without human demonstrations. *Nature*, 625(7995):476–482, 2024. 2

Silviu-Marian Udrescu and Max Tegmark. Ai feynman: A physics-inspired method for symbolic regression. In *Advances in neural information processing systems*, volume 33, pp. 6706–6717, 2020. 1, 2

Hanchen Wang, Tianfan Fu, Yuanqi Du, Wenhao Gao, Kexin Huang, Ziming Liu, Payal Chandak, Shengchao Liu, Peter Van Katwyk, Andreea Deac, et al. Scientific discovery in the age of artificial intelligence. *Nature*, 620(7972):47–60, 2023. 2

Xiaoxue Wang, Fali Wang, Weinan Zhang, Qingyun Wu, Zongyu Wu, Lilin Wang, Ying Wei, Gao Huang, and Yilong Yin. In-context symbolic regression: Leveraging large language models for functional discovery. *arXiv preprint arXiv:2404.19094*, 2024. 2

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. Emergent abilities of large language models. *Transactions on Machine Learning Research*, 2022. ISSN 2835-8856. URL https://openreview.net/forum?id=yzkSU5zdwD. Survey Certification,Outstanding Certification. 1

Jiasheng Ye, Peiju Liu, Tianxiang Sun, Jun Zhan, Yunhua Zhou, and Xipeng Qiu. Data mixing laws: Optimizing data mixtures by predicting language modeling performance. *arXiv preprint arXiv:2403.16952*, 2024. 2, 6

Xinyue Zeng, Haohui Wang, Junhong Lin, Jun Wu, Tyler Cody, and Dawei Zhou. Lensllm: Unveiling fine-tuning dynamics for llm selection. *arXiv preprint arXiv:2505.03793*, 2025. 2, 6

Zhengkang Zhang. Neural scaling laws from large-n field theory: solvable model beyond the ridgeless limit. *Machine Learning: Science and Technology*, 6(2):025026, 2025. 2

Yizhen Zheng, Huan Yee Koh, Jiaxin Ju, Anh T. N. Nguyen, Lauren T. May, Geoffrey I. Webb, and Shirui Pan. Large language models for scientific discovery in molecular property prediction. *Nature Machine Intelligence*, 2025. doi: 10.1038/s42256-025-00994-z. 2

Jinghui Zhong, Junlan Dong, Wei-Li Liu, Liang Feng, and Jun Zhang. Recent advances in symbolic regression. *ACM Computing Surveys*, 57(11), 2025. doi: 10.1145/3735634. 2