

Supplementary Information for the Generative Discovery of Partial Differential Equations by Learning from Math Handbooks

Hao Xu^{1,2}, Yuntian Chen^{1,3,*}, Rui Cao^{4,5}, Tianning Tang^{6,7}, Mengge Du⁸, Jian Li³,
Adrian H. Callaghan⁵, and Dongxiao Zhang^{1,9,*}

¹ Zhejiang Key Laboratory of Industrial Intelligence and Digital Twin, Eastern Institute of Technology, Ningbo, Zhejiang 315200, P. R. China

² Department of Electrical Engineering, Tsinghua University, Beijing 100084, P. R. China

³ Ningbo Institute of Digital Twin, Eastern Institute of Technology, Ningbo, Zhejiang 315200, P. R. China

⁴ College of Oceanic and Atmospheric Sciences, Ocean University of China, Qingdao 266100, P. R. China

⁵ Department of Civil and Environmental Engineering, Imperial College London, London, SW7 2AZ, United Kingdom

⁶ Department of Engineering Science, University of Oxford, Parks Road, Oxford, OX1 3PJ, United Kingdom

⁷ Department of Mechanical and Aerospace Engineering, University of Manchester, Manchester, M13 9PL, United Kingdom

⁸ College of Engineering, Peking University, Beijing 100871, P. R. China

⁹ Institute for Advanced Study, Lingnan University, Tuen Mun, Hong Kong

* Corresponding authors

Email address: ychen@eitech.edu.cn (Y. Chen); dzhang@eitech.edu.cn (D. Zhang)

1. Supplementary text

1.1 Information on the dataset utilized in the experiments

In the discovery of canonical PDEs, eight PDEs from different fields are employed to demonstrate the performance of the proposed generative framework. The information of these PDEs and their respective datasets are provided in Table S5. For evaluating the performance under sparse data and noise, the Allen–Cahn equation is used. Its dataset consists of grid data of 256 spatial observation points in the domain $x \in (-1,1)$ and 201 temporal observation points in the domain $t \in [0,10]$; thus, the data size is 51,456. In the discovery process of the above PDEs, an ANN consisting of five hidden layers, each with 50 neurons, is utilized to construct the surrogate model. 400 population candidates are generated, and the optimization epochs is 5. The learning rate of EqGPT is set to 1×10^{-5} , and a sparsity regularization coefficient $\alpha_0=0.2$ is applied.

For the discovery of PDEs in complex regions, Poisson’s equation has been discovered in the dataset on irregular regions. Here, these datasets are introduced in detail. The simple disk region with a radius of $r=1.5$ is illustrated in Fig. 4a. The Dirichlet boundary condition is $u(x,y)=x\sin(xy)$. The solution is obtained from the simulation via *Mathematica*. The data are collected according to polar coordinates, where $r \in [0.001,1.4926]$ with 200 observation points and where $\theta \in [0,2\pi]$ with 201 observation points. Here, coordinates in the Cartesian coordinate system can be obtained by $x=r\cos(\theta)$, and $y=r\sin(\theta)$. Therefore, there are 40,200 data points in the dataset.

For the “smiley face region” investigated in this work, a region of the smiley face is carved out of the disk region with a radius of 2. This region is defined by the implicit region in *Mathematica*. In this region, the Dirichlet boundary condition is $u(x,y)=\sin(x+y)$. The solution is obtained from the simulation using *Mathematica*. The data are collected in a rectangular region in $x \in [-4,4]$ with 250 observations and $y \in [-4,4]$ with 250 observations. Here, the observation data at the empty locations are deleted. Therefore, there are 44,711 valid data points in the dataset.

For the region of “EITech”, which consists of discontinuous glyph boundaries, the Dirichlet boundary condition, $u(x,y)=0$, is adopted for each alphabet. This region is defined by the *boundaryDiscretizeGraphics* function in *Mathematica*, and the solution is obtained from the simulation by *Mathematica*. The data are collected in a rectangular region in $x \in [4.3,141]$ with 800 observations and $y \in [19.6,54.6]$ with 200 observations. Here, the observation data at the empty locations are deleted. Therefore, there are 48,367 valid data points in the dataset.

For the 3-dimensional (3D) space shuttle region, the Dirichlet boundary condition is as follows: $u(x,y,z)=1$ if $z \leq 1.3$ and $u(x,y,z)=0$ if $x \leq -7$. This region is defined by the *boundaryDiscretizeGraphics* function in *Mathematica*, and the solution is obtained from the simulation by *Mathematica*. The data are collected in a rectangular region with $x \in [7.65,7.04]$ with 300 observations, $y \in [-4.68,4.68]$ with 200 observations, and $z \in [-1.35,4.16]$ with 100 observations. Here, the observation data at the empty locations are deleted. Therefore, there are 438,960 valid data points in the dataset. In the discovery process, the settings are similar to those described above while the sparsity

regularization coefficient $\alpha_0=1$.

For the vibration of an H-shaped elastic membrane, the initial condition is $u(0, x, y) = 2e^{-125(x-0.25)^2 + (y-0.5)^2}$, and $u_t(0, x, y) = 0$. The boundary condition is the Dirichlet boundary, $u(t, x, y)=0$. This region is defined by the *RegionDifference* function in *Mathematica*, which is carved out of the rectangular region where the length of x is 2 and the length of y is 1. The excavated parts are $x \in [0.9, 1.1] \cup y \in [0, 0.4]$ and $x \in [0.9, 1.1] \cup y \in [0.6, 1.0]$. The solution is obtained from the simulation using *Mathematica*. The data are collected in a rectangular region in $x \in [0, 2]$ with 200 observations, $y \in [0, 1]$ with 100 observations, and $t \in [0, 2]$ with 200 observations. Here, the observation data at the empty locations are deleted. Therefore, there are 3,774,981 valid data points in the dataset. In the discovery process, the settings are similar to those described above, while 800 population candidates are generated and the sparsity regularization coefficient $\alpha_0=1$ is applied.

For the oil-water two phase flow in 3-dimensional space, the data are generated from the Unconventional Oil and Gas Simulator (UNCONG) under the assumptions of incompressibility and negligible capillary pressure. The simulation domain consists of a three-dimensional discretized geological formation, represented by a structured grid. Two injection wells are positioned at the upper-left and lower-left corners of the grid, while a single production well is located at the center of the right boundary. From the initial day of simulation, water is continuously injected through the injection wells to displace oil within the reservoir and promote its migration toward the production well. The spatial grid is defined over the domain $[x, y] \in [0, 30]$ with a uniform spacing of $\Delta x = \Delta y = 1\text{m}$, and $z \in [0, 8]$ with $\Delta z = 1\text{m}$. Temporal discretization spans $t \in [0, 30]$ with a time step of $\Delta t = 1$ day. The simulation yields the spatial-temporal distribution of the water-phase pressure field P_w , saturation field S_w , and effective permeability field K_w and K_o , each represented by 600,000 grid-based data points. Finite difference is conducted on the grid-data for differentiation. During PDE discovery, potential differential terms associated with S_w , K_w , and K_o , such as $\partial S_w / \partial t$, ∇K_w , ∇K_o , ΔK_w , and ΔK_o , are incorporated into the vocabulary, while terms related to P_w are constructed analogously to the single-variable setting. The first token is fixed to be $\partial S_w / \partial t$. The symmetry in the x and y directions is used to constrain the generated equations. 600 population candidates are generated, and the optimization epochs is 5. The learning rate of EqGPT is set to 1×10^{-5} , and a sparsity regularization coefficient $\alpha_0=0.1$ is applied.

For the discovery of previously undisclosed governing equations for highly nonlinear surface gravity waves propagating toward breaking, the data come from real-world wave-tank experiments. A total of 12 independent experiments are conducted with different initial conditions of the wave paddle, which is displayed in Table S4. The spatial surface elevation recorded in each experiment was reconstructed at different time instants before the onset of breaking by employing the image-processing technique. For each experiment, the recorded time period spans between 40 and 100 frames. Each frame contains approximately 4,800 spatial points, covering three non-contiguous spatial subdomains $x \in [8.1516, 9.36] \cup [9.74, 10.95] \cup [11.392, 12.574]$. Data from 12

individual experiments are aggregated for joint PDE discovery. For each experiment, a surrogate model is constructed using an artificial neural network (ANN) comprising six hidden layers with 60 neurons per layer. Each network is trained for up to 50,000 iterations. The input is non-dimensional variable x^* and t^* , and the output is the non-dimensional surface elevation η^* . Automatic differentiation is employed to compute spatial and temporal derivatives from the trained surrogate models. During the PDE discovery process, a population size of 400 candidate equations is used, with five optimization epochs. The learning rate for the EqGPT model is set to 1×10^{-5} , and a sparsity regularization coefficient $\alpha_0=0.02$ is applied in the reward function to encourage parsimonious expressions. For each generated PDE, the least squared regression is performed on the data from each of the 12 experiments independently. The individual rewards from these experiments are then averaged to obtain a global reward score, which is used to guide the optimization of the generative model.

1.2 Comparison between EqGPT learning from the dataset with and without the target PDE

In this work, to guarantee the validity of the experiments, the target PDE is temporally deleted from the dataset, and the EqGPT generative model is retrained by the modified dataset. This scenario presents a higher level of complexity, closely resembling real-world situations in which the underlying governing equations remain undiscovered. Notably, in practical applications, there is also the possibility that the underlying governing equations align with existing PDE systems, albeit under various conditions. Therefore, the EqGPT trained on the entire PDE dataset can handle practical scenarios. Here, a comparison between the discovery of PDEs with EqGPT trained on the dataset with and without the target PDE is conducted. Similarly, 8 canonical PDEs are investigated. Here, 10 independent experiments with different random seeds are conducted when discovering each PDE, and the success rate is defined as:

$$S_r = n_{\text{success}} / n_{\text{total}} , \quad (\text{S.1})$$

where, S_r refers to the success rate; n_{success} refers to the number of times the correct equation form has been successfully discovered; and n_{total} refers to the number of experiments in total (10 in this work). The results are provided in Fig. S2. The figure shows that the success rate with the target equation (red line) is slightly higher than that without the target equation (blue line). However, even when trained without a target equation, EqGPT achieves success rates above 0.7 in all of the experiments, with minimal to no adverse effects in most instances. This finding demonstrates that the proposed method maintains robust stability under challenging conditions. Notably, it is found that the Chaffee–Infante equation is most affected, suggesting that generating the term u^3 may be particularly challenging without learning the target PDE form.

The average discovered epochs with and without the target PDE are also provided in Fig. S2; it is measured by the average of the epochs for which the PDE is first discovered in all 10 independent experiments, which can reflect the optimization efficiency of the method. Notably, for the cases in which the PDE cannot be discovered

successfully, the number of epochs is regarded as 5. The figure shows that learning the target form does not strongly affect the optimization efficiency since, in both situations, the average number of discovered epochs is lower than 3, which indicates high speed. Interestingly, the average number of discovered epochs even decreases without learning the target form in some cases, such as the convection diffusion equation and the KdV equation. Similarly, the Chaffee–Infante equation is most affected, which shows an increase of nearly 1 epoch when the target PDE is not learned.

1.3 Performance of PDE discovery with different numbers of PDE training EqGPT

In this work, the EqGPT model is trained on the established PDE dataset, which consists of 221 PDEs. To examine the potential of EqGPT, different numbers of PDEs are utilised in the dataset to train the EqGPT model, which is employed to discover the PDEs. Here, the KdV equation, Eq.(6.2.12), and Eq. (8.14.1d) are adopted, where 10,000 data points with no noise are randomly selected to train the surrogate model. Moreover, different numbers of PDEs in the dataset, including 0, 5, 10, 25, 50, 100, 200, and 220 PDEs, are randomly selected to train the EqGPT generative model. Notably, the target PDE has been deleted from the dataset. 10 independent experiments are conducted with different random seeds to calculate the success rate. The results are provided in Fig. S3. The success rate clearly presents an increasing trend with the number of PDEs training the EqGPT. When the generative model is not trained with any PDEs, it is in its initial state and is unable to identify the correct PDE. However, upon training with even a small set of 5 PDEs, EqGPT’s success rate improves to 0.3 for the KdV equation. Meanwhile, discovering PDEs with more intricate or irregular terms requires exposure to a broader set of equation forms during training. Notably, as the number of PDEs in the training dataset increases, the success rate also increases, highlighting the potential of the proposed framework; this suggests that a more extensive PDE dataset in the future could be instrumental in tackling more complex problems. Notably, training EqGPT with the current PDE dataset achieves a success rate of over 0.8, indicating a commendable level of performance.

1.4 Discovery of PDEs with equivalent forms

In practice, many conservation laws and other physically meaningful transformations lead to PDEs with mathematically equivalent formulations. To examine how the proposed EqGPT framework addresses the issue of such mathematical equivalences among PDE terms, we designed an experiment with a compound-form PDE, denoted as PDE_compound:

$$u_t - 0.2(uu_x)_x = 0. \quad (\text{S.2})$$

This PDE does not appear in the established PDE dataset, and thus poses a challenge for the discovery task. The observation dataset is obtained from the simulation via *Mathematica*. The initial condition is $u(0,x)=\sin(\pi x)$, and the boundary condition is $u(t,0)=u(t,1)=0$. The dataset contains the grid data of 200 spatial observation points in

the domain $x \in [0,1]$ and 200 temporal observation points in the domain $t \in [0,1]$; thus, the data size is 40,000. Here, 10,000 data points with no noise are randomly selected to train the surrogate model. The EqGPT successfully identified the correct PDE form in the 3rd optimization epoch, though with relatively lower coefficient accuracy due to the rarity and complexity of this structure. Importantly, beyond the compound expression (Eq. S.2), the framework also discovered its mathematically expanded equivalent:

$$u_t - 0.2uu_{xx} - 0.2u_x^2 = 0. \quad (\text{S.3})$$

This result demonstrates that the proposed framework is capable of independently recognizing and recovering both compound and expanded representations of the same PDE, thereby exhibiting flexibility in handling mathematically equivalent formulations.

1.5 The performance of the proposed method under sparse and noisy data

In this section, the performance of the proposed method is examined by discovering the Allen–Cahn equation under different numbers of data points and different levels of noise. The true equation form is written as:

$$u_t - 0.003u_{xx} - u + u^3 = 0. \quad (\text{S.4})$$

This PDE is comparatively difficult to discover, as it comprises four terms and features a minor coefficient for the diffusion term (Fig. S5a). Similarly, the PDE discovery process is carried out on the meta-data generated by the surrogate model trained from observation data. As illustrated in Fig. S5b and S5c, it is evident that the surrogate model is capable of accurately reconstructing the physical field from highly noisy and sparse data. The settings of these algorithms are detailed in the Methods.

Fig. S5d presents the coefficient errors when using the Allen–Cahn equation under different numbers of data points and different levels of noise. In this context, the red blocks indicate instances in which the algorithm was unable to identify the correct PDE structure, whereas the blue blocks denote successful discoveries. A deeper blue colour indicates a lower coefficient error. The figure reveals that our proposed generative framework achieves satisfactory accuracy and efficient optimization, typically requiring only 1 to 3 epochs. This efficiency is attributed to the learned knowledge of PDE forms in books, which enables the model to generate proper structures for optimization.

Supplementary Tables and Figures

Table S1. The tokens used for constructing PDEs in the EqGPT framework

Index	Token	Index	Token	Index	Token
1	E	20	u_{xxx}	39	u_z
2	+	21	u^3	40	u_{zz}
3	*	22	x	41	Δu_{tt}
4	/	23	u_{yy}	42	$x+y$
5	S	24	$(1/u)_{xx}$	43	e^x
6	u_t	25	$(u_x/u^2)_x$	44	u_{yyt}
7	u	26	u_{xxxxx}	45	$\sin(t)$
8	u_x	27	$(u_t)^3$	46	$\sin(x)$
9	u_{xx}	28	\sqrt{u}	47	$(u_{xx}+u_x/2)^2$
10	u_{xxt}	29	$\sin(u)$	48	x^4
11	$(u_t)^2$	30	$\sinh(u)$	49	\sqrt{x}
12	u_{xt}	31	$\Delta^2 u$	50	e^{-y}
13	u_{tt}	32	u_y	51	t
14	$(u_x)^2$	33	Δu	52	u_{yyy}
15	u_{xxx}	34	y	53	$(uu_x)_t$
16	$(uu_x)_x$	35	x^2	54	$(uu_x)_{xx}$
17	u_{xxtt}	36	y^2	55	$(u^3)_{xx}$
18	$(u^4)_{xx}$	37	$(u_y)^2$	56	$(u(u^2)_{xx})_{xx}$
19	u^2	38	u_{xy}		

Table S2. The top-5 equations in each epoch of the optimization process when discovering KdV equation with 1000 data

	Rank 1	Rank 2	Rank 3	Rank 4	Rank 5
Epoch 1	$uu_x, u_t, u^2 u_x,$ u_{xxx}	$uu_x, u_t, u/t,$ u_{xxx}	$u_{xxx}, u_x u, (u_x)^2 u^2,$ u_t	$(uu_x)_x, u_{tt},$ u_{xxxx}, u_{xx}	$u_{xxx}, uu_x, u^2 u_x,$ $(u(u^2)_{xx})_{xx}, u_{xx}, u_t$
Epoch 2	$uu_x, u_t, u^2 u_x,$ u_{xxx}	$uu_x, u_{xxx}, u_t,$ $(u_x)^2$	$u_x u, u_{xxx}, u_{xx},$ u_t	$uu_x, u_t, u/t,$ u_{xxx}	$u_{xxx}, u_x u,$ $(u_x)^2 u^2, u_t$
Epoch 3	uu_x, u_t, u_{xxx}	$u_{xxx}, u_t, u^2 u_x,$ u_{xxx}	$u_x u, u_{xxx}, u_t,$ $(u_x)^2$	$u_x u, u_{xxx},$ u_{xx}, u_t	$uu_x, u_t, u/t,$ u_{xxx}
Epoch 4	uu_x, u_t, u_{xxx}	$uu_x, u_t, u^2 u_x,$ u_{xxx}	$u_{xxx}, u_x u^2, u_t,$ $u_x u$	$u_x u, u_x u_{xxxx},$ u_t, u_{xxx}	uu_x, u_t, u_{xxx}, u_x
Epoch 5	uu_x, u_t, u_{xxx}	u_{xxx}, uu_x, u_t	$u_x u, u_t, u^2 u_x,$ u_{xxx}	$u_{xxx}, (uu_x)_x,$ uu_x, u_t	$u_{xxx}, u_x u^2, u_t, uu_x$

Table S3. The discovered equation of 2-dimensional (2D) Burgers' equation under different noise levels.

Noise level	Discovered equation
0%	$u_t + 1.01uu_x + 1.01uu_y - 0.0097\Delta u = 0$
1%	$u_t + 1.00uu_x + 1.02uu_y - 0.0097\Delta u = 0$
5%	$u_t + 1.02uu_x + 1.01uu_y - 0.0096\Delta u = 0$
10%	$u_t + 0.99uu_x + 1.03uu_y - 0.0096\Delta u = 0$
15%	$u_t + 1.87uu_x + 2.07uu_y - 0.0093\Delta u = 0$
20%	$u_t + 2.19uu_x + 2.30uu_y - 0.0021\Delta u - 58.7u \cdot u_{tt} = 0$

Table S4. The initial conditions of the wave paddle for conducting 12 independent experiments of wave-breaking. Here, G is the peak enhancement factor, T_p is the peak wave period of the JONSWAP-type spectra, and A is the total amplitude summed across all underlying wave components. C_1 , C_2 , and C_3 are the regressed coefficients of the discovered equation.

Case serial number	G	T_p	A	C_1	C_2	C_3
1	2	1.2	80	1.461	1.936×10^{-3}	-7.38×10^{-5}
2	2	1.2	90	1.38	7.779×10^{-4}	-3.89×10^{-5}
3	2	1.2	100	1.36	1.091×10^{-3}	-4.63×10^{-5}
4	2	1.2	105	1.426	1.875×10^{-3}	-9.12×10^{-5}
5	2	1.3	90	1.489	1.873×10^{-3}	-5.61×10^{-5}
6	2	1.3	96	1.404	1.320×10^{-3}	-4.25×10^{-5}
7	2	1.3	105	1.469	1.529×10^{-3}	-6.40×10^{-5}
8	2	1.3	130	1.371	1.096×10^{-3}	-5.39×10^{-5}
9	3	1.2	83	1.575	2.376×10^{-3}	-7.75×10^{-5}
10	3	1.2	95	1.511	1.967×10^{-3}	-7.45×10^{-5}
11	3	1.2	100	1.44	1.508×10^{-3}	-7.10×10^{-5}
12	3	1.2	117	1.386	4.819×10^{-4}	1.26×10^{-5}

Table S5. The spatial and temporal domains and data points of the dataset for the canonical PDEs utilized in this work.

Equation name	Equation form	Spatial domain and points	Temporal domain and points
KdV equation	$u_t + uu_x + 0.0025u_{xxx} = 0$	$x \in [-1, 1)$ $n_x=512$	$t \in [0, 1]$ $n_t=201$
Burgers' equation	$u_t + uu_x - 0.1u_{xx} = 0$	$x \in (-8, 8)$ $n_x=256$	$t \in (0, 10]$ $n_t=201$
Convection-diffusion equation	$u_t + u_x - 0.25u_{xx} = 0$	$x \in [0, 2]$ $n_x=256$	$t \in [0, 1]$ $n_t=100$
Chaffee-Infante equation	$u_t - u_{xx} + u - u^3 = 0$	$x \in [0, 3]$ $n_x=301$	$t \in (0, 0.5)$ $n_t=200$
Wave equation	$u_{tt} - u_{xx} = 0$	$x \in [0, \pi]$ $n_x=161$	$t \in [0, 2\pi]$ $n_t=321$
KG equation	$u_{tt} - 0.5u_{xx} + 5u = 0$	$x \in [-1, 1)$ $n_x=201$	$t \in [0, 3]$ $n_t=201$
Eq. (6.2.12) *	$0.1u_{xt} + u_t + 0.1u_x = 0$	$x \in [0, 5]$ $n_x=500$	$t \in [0, 10]$ $n_t=500$
Eq. (8.14.1d) *	$u_t + u_x/x + 0.25u_{xx} = 0$	$x \in [1, 2]$ $n_x=100$	$t \in [0, 1]$ $n_t=251$
PDE_compound	$u_t - 0.2(uu_x)_x = 0$	$x \in [0, 1]$ $n_x=200$	$t \in [0, 1]$ $n_t=100$
Eq. (7.2.12) *	$-0.1u_{xxt} + u_t + 0.1u_x = 0$	$x \in [0, 5]$ $n_x=500$	$t \in [0, 10]$ $n_t=500$
Eq. (6.12.28) *	$xu_{tt} - 0.1u_{xx} = 0$	$x \in [0, 5]$ $n_x=500$	$t \in [0, 10]$ $n_t=500$
Fujita–Storm equation	$u_t - 0.05(u_x/u^2)_x = 0$	$x \in [0, 5]$ $n_x=500$	$t \in [0, 5]$ $n_t=500$

*The equation name corresponds to the index of the math handbook, *Nonlinear Partial Differential Equations for Scientists and Engineers*²⁵.

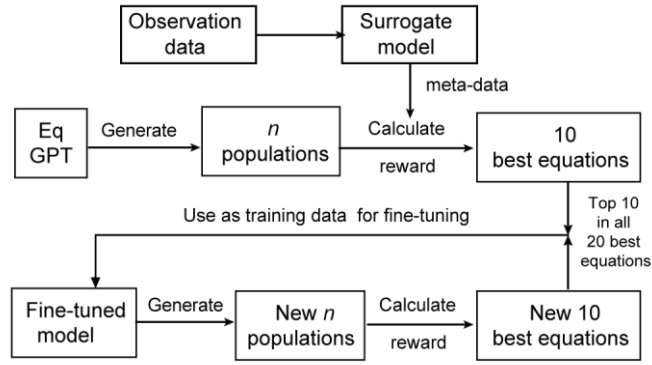


Fig. S1. Flow chart of the optimization cycle for discovering PDEs. The workflow shows the generation–evaluation–optimization cycle. EqGPT first generates n candidate equations, whose performance is evaluated by a surrogate model trained on observation data. The top equations are selected and used to fine-tune EqGPT, which then produces new populations in the next cycle. Repeated generation–evaluation–optimization progressively improves equation quality, yielding concise and accurate PDEs that best capture the system’s underlying dynamics.

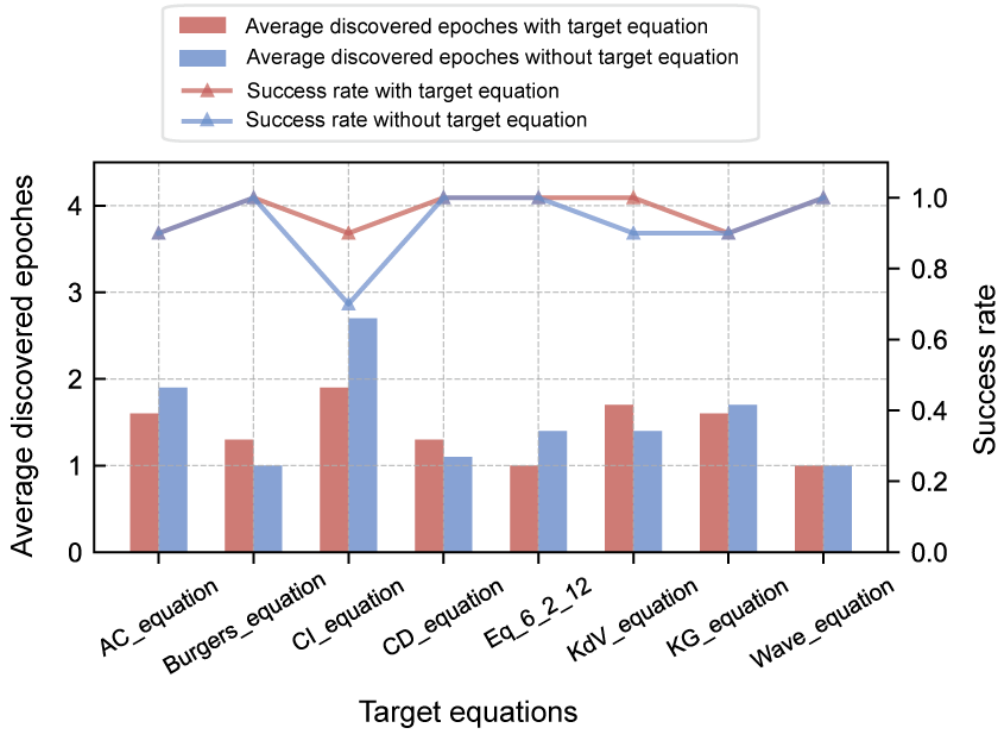


Fig. S2. Comparison between training EqGPT with and without the target equation. Eight canonical PDEs are investigated, including the Allen Cahn (AC) equation, Burgers’ equation, the Chaffee-Infante (CI) equation, the convection diffusion (CD) equation, Eq. (6.2.12) in the handbook, the Korteweg-De Vries (KdV) equation, the Klein–Gordon (KG) equation, and the wave equation. The bar plots show the average number of discovered epochs. The line charts denote the success rate.

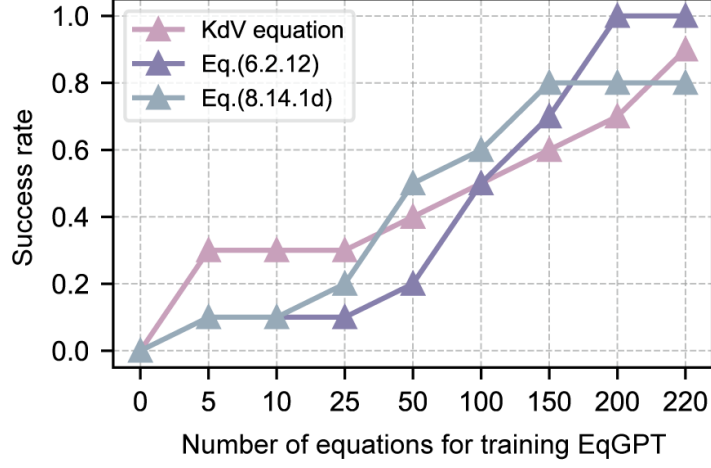


Fig. S3. Discovery of canonical PDEs with different numbers of equations for training EqGPT. This illustrates how the number of training equations influences EqGPT’s ability to rediscover canonical PDEs, including the KdV equation and two benchmark equations from PDE datasets. As the training set expands, the success rate steadily increases. When trained with more than 150 equations, EqGPT achieves near-perfect recovery of all target PDEs, demonstrating the effectiveness of knowledge guidance.

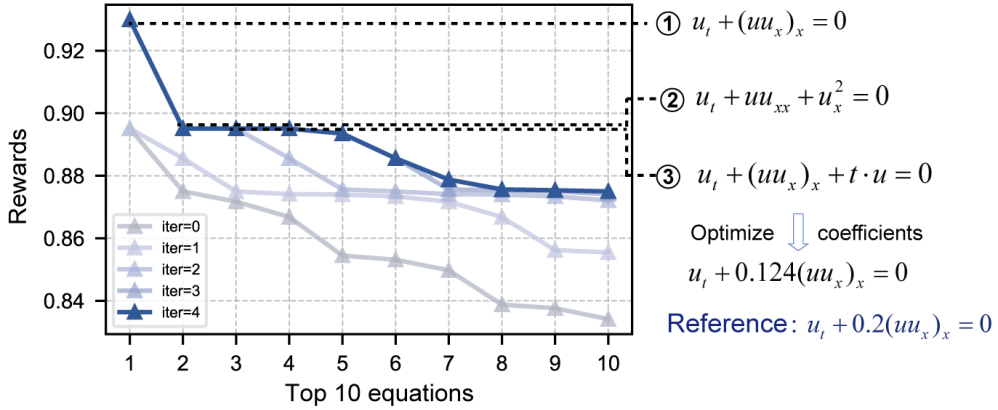


Fig. S4. Discovery of PDEs with equivalent forms. The rewards of the top 10 equations generated from the EqGPT model in each optimization epoch (**left**), and the discovered PDE and the true PDE (**right**). The best-discovered form and the suboptimal form are mathematically equivalent.

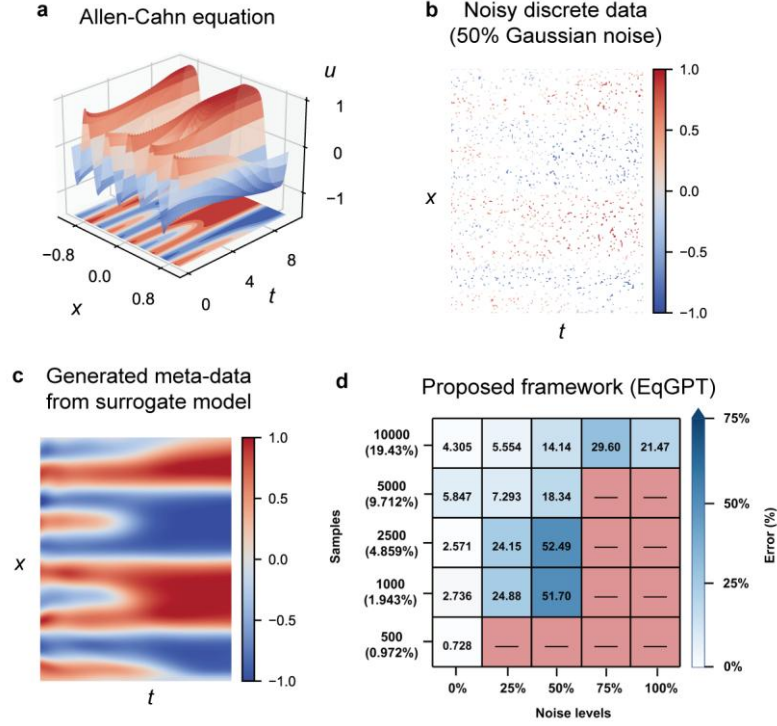


Fig. S5. The performance of the proposed method under sparse and noisy data when discovering the Allen-Cahn equation. (a) Illustration of the dataset for the Allen-Cahn equation. (b) Visualisation of noisy discrete data with 50% noise. (c) The reconstructed metadata from the surrogate model trained on observation data. (d) The coefficient error for discovering the Allen-Cahn equation under different levels of noise and data points. The red blocks indicate instances in which the algorithm was unable to identify the correct PDE structure, whereas the blue blocks denote successful discoveries. A deeper blue colour corresponds to a lower coefficient error.