Stony Brook University

From Prediction to Creation: Transforming Scientific Discovery with Artificial Intelligence

Presenter: Wenhan Gao

Advisor: Yi Liu

Department of Applied Mathematics and Statistics

FAR BEYOND

Stony Brook University

About Me

- Now: 4th year Ph.D. student in Applied Mathematics supervised by Professor Yi Liu
- Past: B.S. from Stony Brook in Pure Mathematics and Applied Mathematics
- Research Areas: AI for Science
  - Generative (Probabilistic) Models
  - Equivariant Neural Networks
  - Large Language Models
- Homepage: https://wenhangao21.github.io/

FAR BEYOND

Talk Outline

- Introduction to AI and Neural Networks
- Introduction to AI4Sci and My Research
- Discriminative Models:
  - AI for Continuum Systems (PDEs)
    - Physics Informed Neural Networks
    - Neural Operators
  - AI for Atomistic Systems
- Generative Models
  - Introduction to Generative Models
  - Diffusion Models for Molecular Generation
  - LLMs for Science
- Acknowledgement

Preliminary: Artificial Intelligence

- Artificial intelligence (AI) focuses on creating systems capable of autonomously performing tasks that typically require human intelligence
  - Recognizing patterns
  - Understanding natural language
  - Reasoning
  - Making decisions

"Ex machina" is the second half of the Latin phrase Deus ex machina, which translates to "God from the machine".

Preliminary: Functions

- Mathematically, a **function** is a rule that assigns **each input** from a set (the *domain*) to **a deterministic output** in another set (the *codomain*).

- Formally:

$$f: X \rightarrow Y$$

  - X: domain (the set of possible inputs)
  - Y: codomain (the set of possible outputs)
  - For every $x \in X$, there exists a unique $y \in Y$ such that $y = f(x)$

Preliminary: Learning Functions

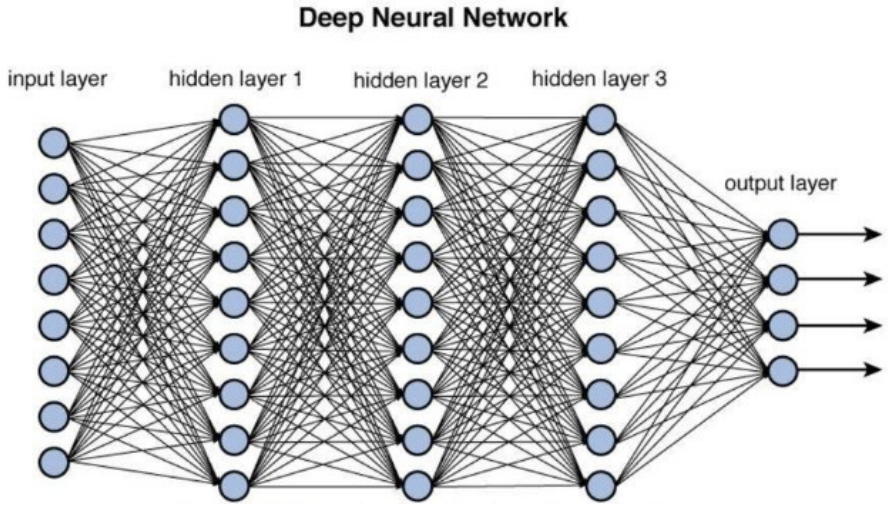
- Why do we care about functions? A lot of tasks can be described as learning functions.
  - Image Classification: Image ↦ Label
  - Image Segmentation: Image ↦ Pixel-wise Labels
  - Machine Translation: Sentence in Language A ↦ Sentence in Language B
- In practice, computers only process **numerical values.**
  - Input: Image → represented as a 3D tensor of pixel values
    - $x \in R^{H \times W \times C}$
  - Output: Class → a probability vector over classes
    - $y \in R^{K}$, then pick $\arg\max(y)$

Preliminary: Deep Neural Networks

- A prominent example of AI systems is the (deep) neural network (NN), which can be thought of as a universal function approximator[1].
- Mathematically, a feedforward neural network is a function composition:

$$f(x) = f^{(L)} \circ f^{(L-1)} \circ \cdots \circ f^{(1)}(x),$$

  - Each f consists of a linear transformation (matrix multiplication and bias) and a nonlinear activation function.



[1] Assuming measurable and under mild regularity conditions
Image source: https://botpenguin.com/glossary/deep-neural-network

Preliminary: Function Basis

- In functional analysis, we often describe functions in terms of a function basis:

$$f(x) = \sum_i c_i \phi_i(x).$$

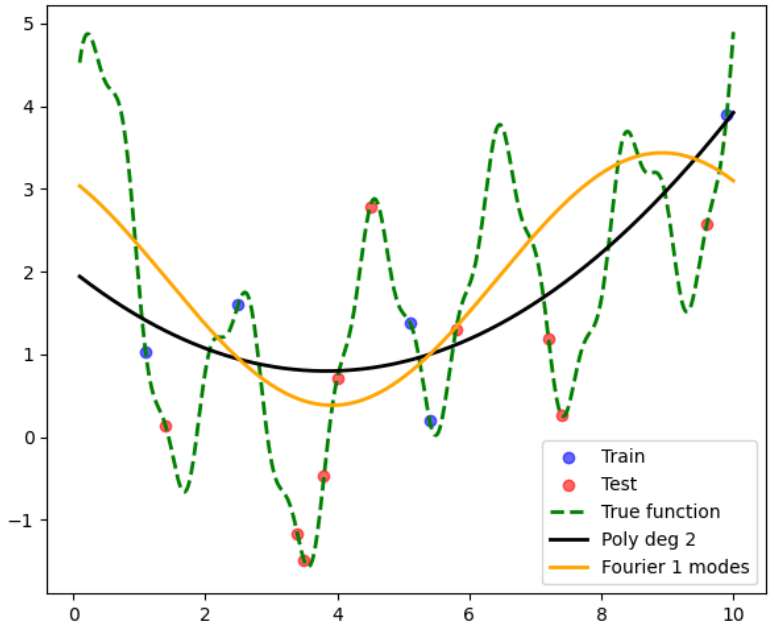
  - *E.g. polynomial expansion, Fourier expansion*
- NN is analogous to having an adaptive/learnable function basis.
  - One layer of a feedforward neural net can be written as

$$f(x) = \sum_{j=1}^{m} \sigma\left(\mathrm{w}_j^\top x + b_j\right)$$

  - The basis functions are $\phi_j(x) = \sigma\left(\mathrm{w}_j^\top x + b_j\right)$.

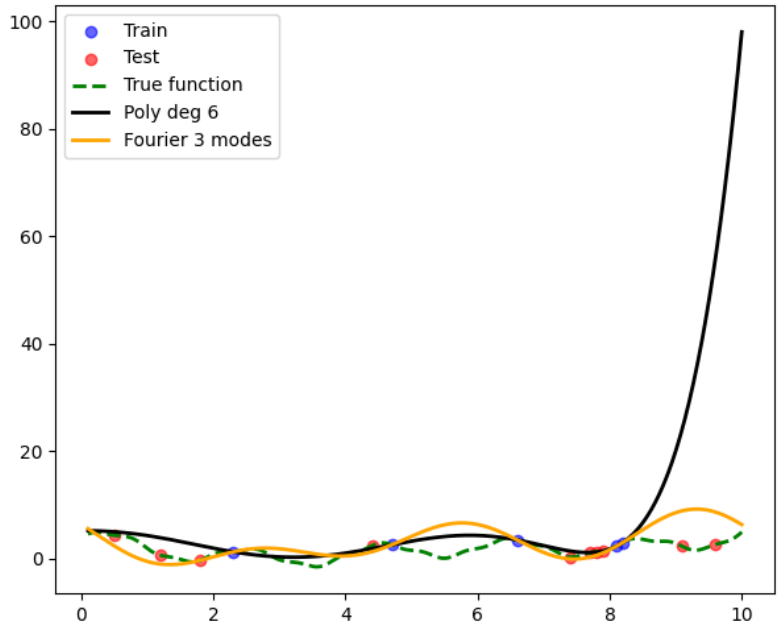
Preliminary: Function Fitting

- Function fitting is the process of finding a function that best describes a set of data points.



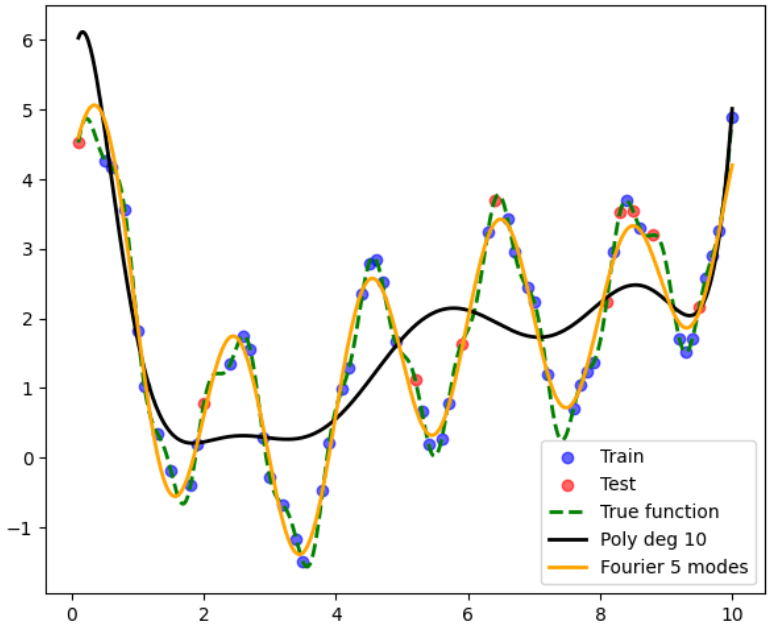
Both Underfitting

Need more basis functions



Polynomial basis: overfitting
Trigonometric basis: a little overfitting

Need more data



Polynomial basis: underfitting
Trigonometric basis: a good fit

A good basis is important

- Functions for many tasks are much more complex!

Preliminary: Learning Representation

- How NN learns ⟶ function fitting with a learnable function basis
  - Training data ⟶ Observation points
  - Width of a Layer ⟶ Number of basis functions (e.g. degree in polynomial fitting)
  - Depth of NN ⟶ Improved function bases through hierarchical basis construction
    - These basis functions can be called features or representations

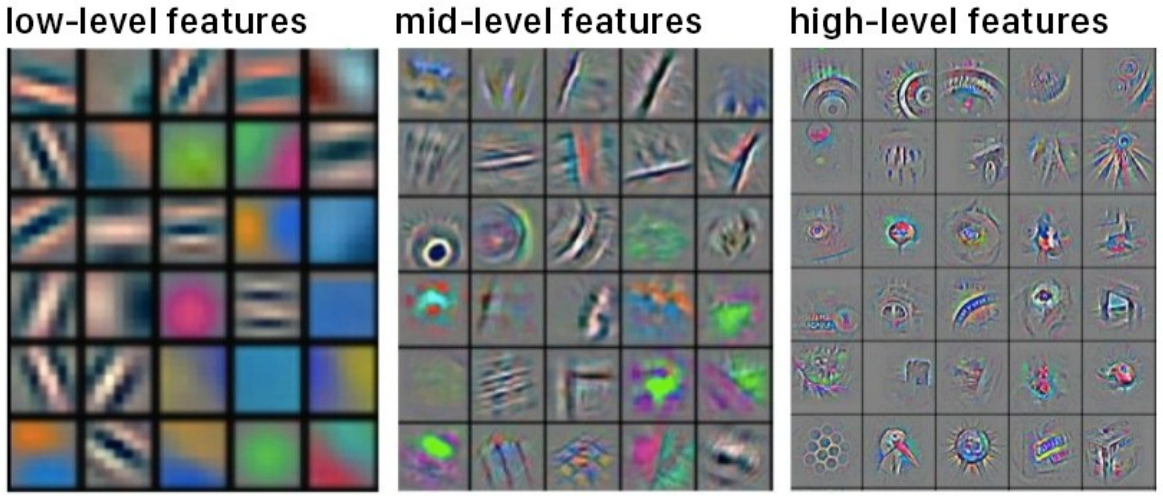


Image source: https://tvirdi.github.io/2017-10-29/cnn/

Preliminary: Challenges with NNs

- Training data ⟶ Observation points
  - We need a sufficient, often large, amount of training data.
- Network designs ⟶ Better function basis
  - We need to design network architectures tailored to the tasks at hand.
- Learned basis ⟶ Black-box representations
  - We need to ensure that the network is trustworthy.

My research solves these challenges: data, design, and trustworthiness.

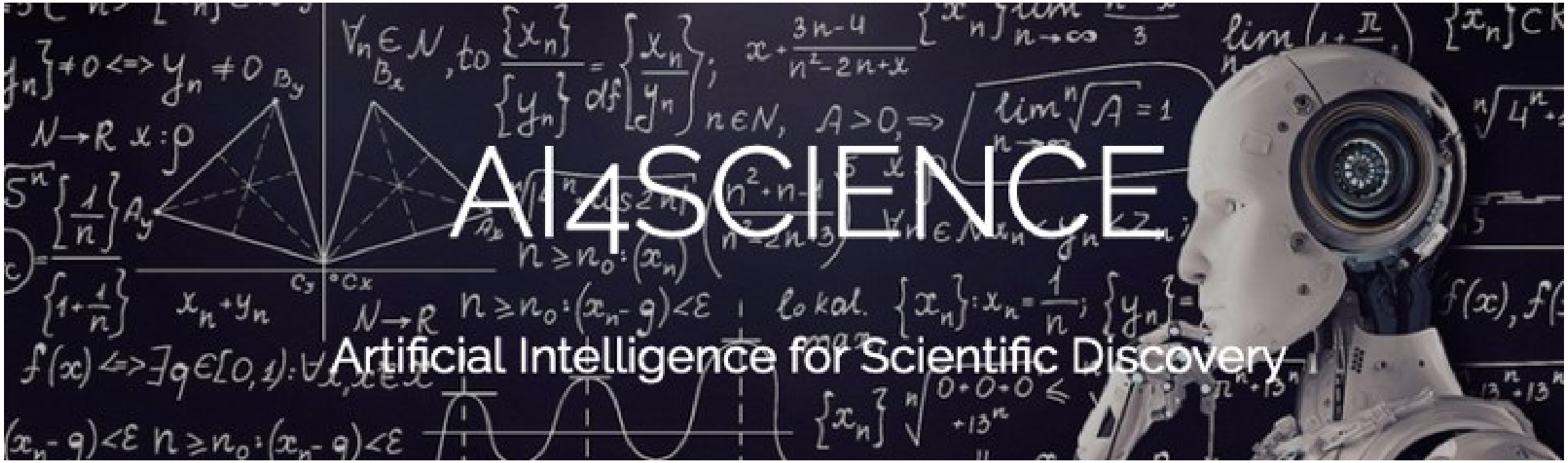
Talk Outline

- Introduction to AI and Neural Networks
- Introduction to AI4Sci and My Research
- Discriminative Models:
  - AI for Continuum Systems (PDEs)
    - Physics Informed Neural Networks
    - Neural Operators
  - AI for Atomistic Systems
- Generative Models
  - Introduction to Generative Models
  - Diffusion Models for Molecular Generation
  - LLMs for Science
- Acknowledgement

Stony Brook University

AI for Science (AI4Sci)

AI4Sci refers to the use of recent advances in artificial intelligence and deep learning to solve problems in sciences: computational chemistry, PDEs, material science, drug design, etc..



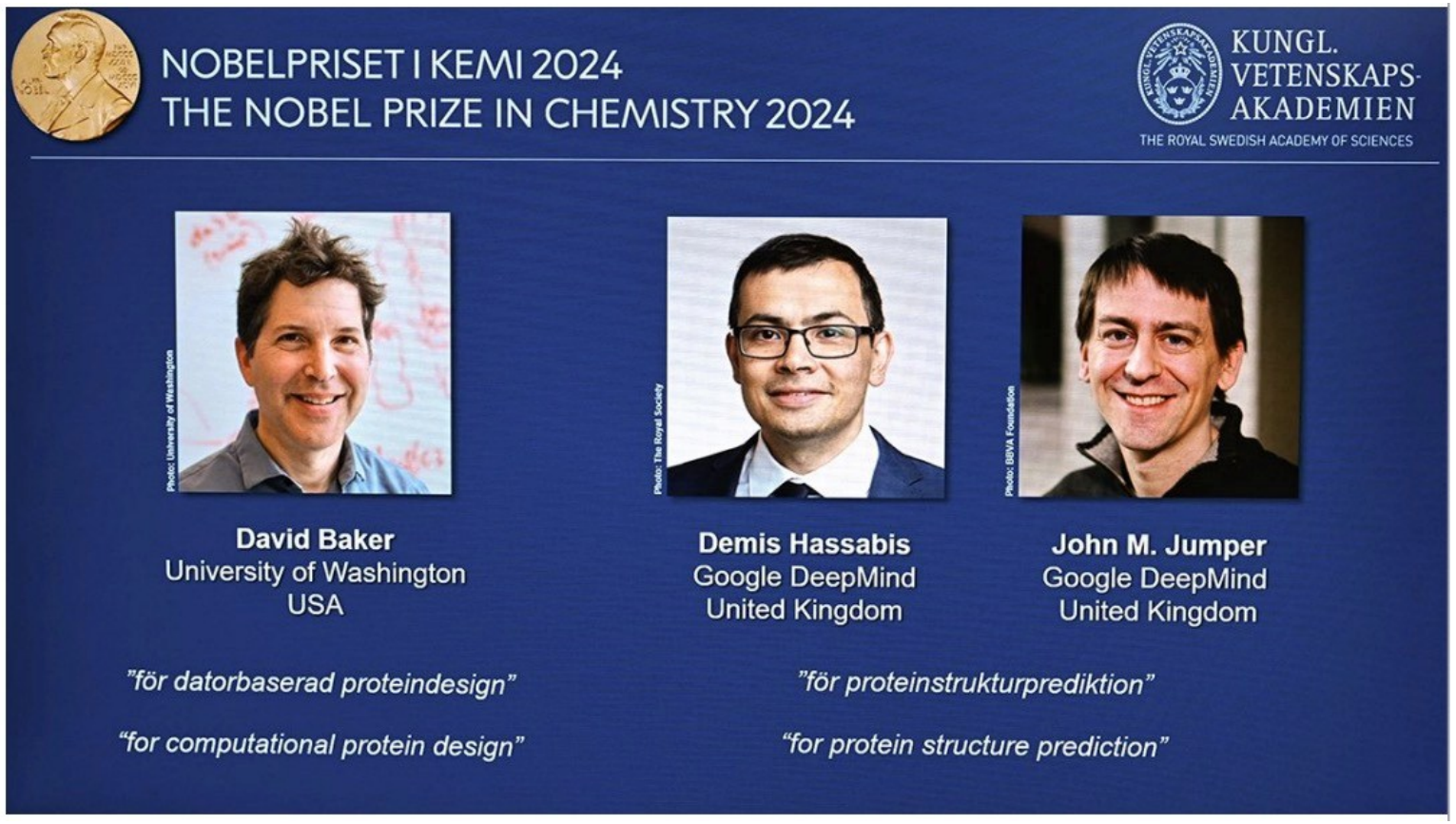
FAR BEYOND

Best Time for AI4Sci

Nobel Prize in Physics → AI

AI pioneers John J. Hopfield and Geoffrey E. Hinton for their contribution to AI and ML

Best Time for AI4Sci

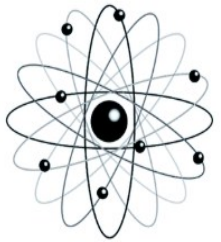


Nobel Prize in Chemistry → AI4Science

Demis Hassabis, and John Jumper for their contribution to AlphaFold - protein structure prediction with AI

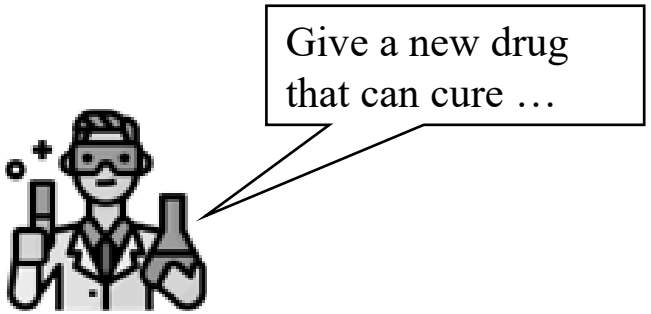
My Research

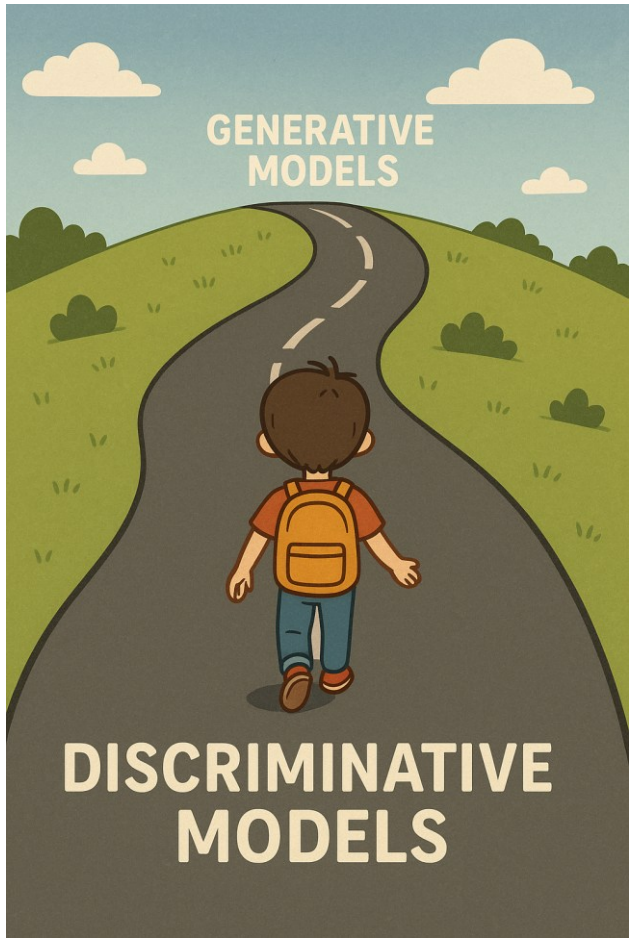
- Discriminative Models:
  Learn patterns from data and **select the best fit**



→ Dipole moment
$\mu = 1.69D$

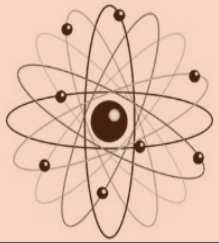
- Generative Models:
  Learn patterns from data and **create new possibilities**





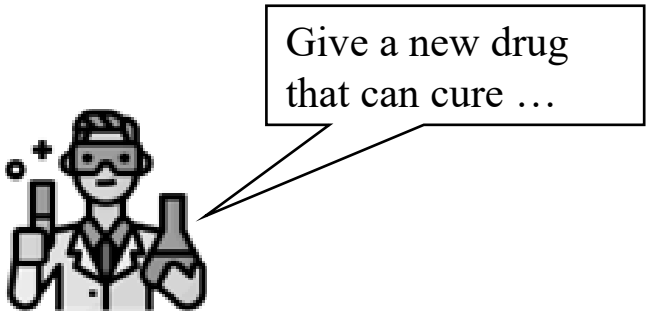
My Research: Discriminative Models

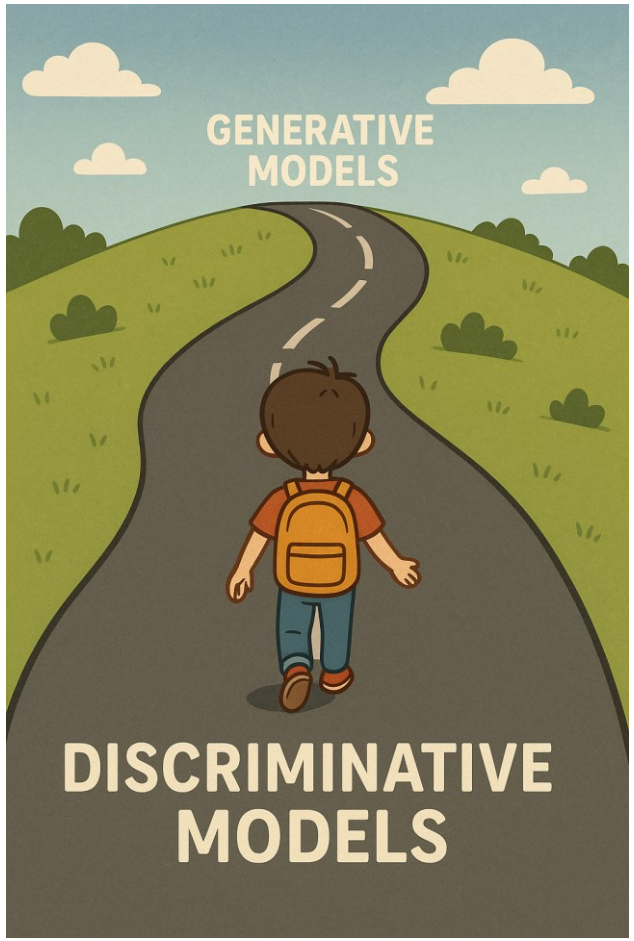
➢Discriminative Models:
Learn patterns from data and **select the best fit**



→ Dipole moment
$\mu = 1.69D$

➢Generative Models:
Learn patterns from data and **create new possibilities**



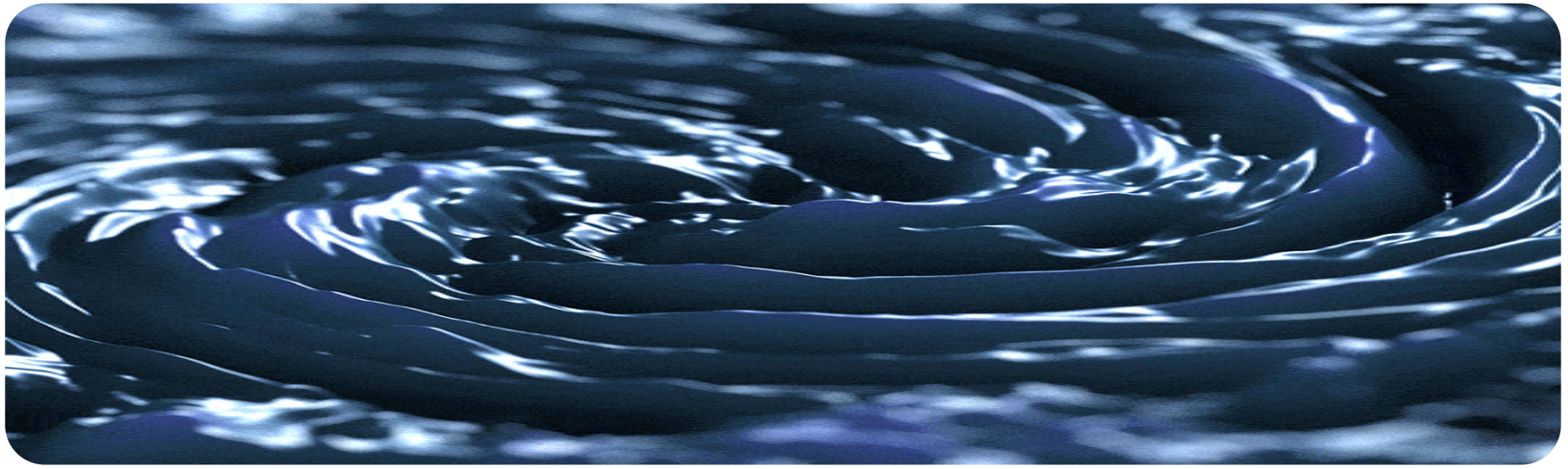


Talk Outline

- Introduction to AI and Neural Networks
- Introduction to AI4Sci and My Research
- Discriminative Models:
  - AI for Continuum Systems (PDEs)
    - Physics Informed Neural Networks
    - Neural Operators
  - AI for Atomistic Systems
- Generative Models
  - Introduction to Generative Models
  - Diffusion Models for Molecular Generation
  - LLMs for Science
- Acknowledgement

Neural PDE Solvers

- A partial differential equation (PDE) models continuum systems through relations among partial derivatives.
- Solving PDEs is central to nearly all scientific and engineering disciplines.



Navier Stokes Equation (Fluid Dynamics)

PINNs (Solving One Instance)

Consider the following general form of a PDE for $u(\boldsymbol{x})$:

$$\begin{cases} \mathcal{D}u(\boldsymbol{x}) = f(\boldsymbol{x}), & \text{in } \Omega, \\ \mathcal{B}u(\boldsymbol{x}) = g(\boldsymbol{x}), & \text{on } \partial\Omega, \end{cases}$$

we wish to approximate $u(\boldsymbol{x})$ with a neural network, denoted by $\phi(\boldsymbol{x};\boldsymbol{\theta})$ the following optimization problem:

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}) := \arg\min_{\boldsymbol{\theta}} \|\mathcal{D}\phi(\boldsymbol{x};\boldsymbol{\theta}) - f(\boldsymbol{x})\|_2^2 + \lambda\|\mathcal{B}\phi(\boldsymbol{x};\boldsymbol{\theta}) - g(\boldsymbol{x})\|_2^2$$

Intuition: We parametrize the solution with a neural network (mesh-free!!!) and penalize the neural network by the extent to which it violates the PDE/boundary/initial conditions (no data needed!!!).

Barron's Theorem → Polynomial Complexity → Solving High-dimensional PDEs

My Contribution

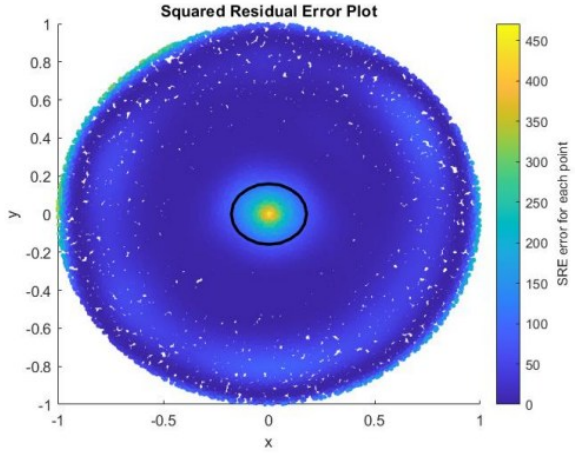
- AL-PINN, JCP 2023 (Data):

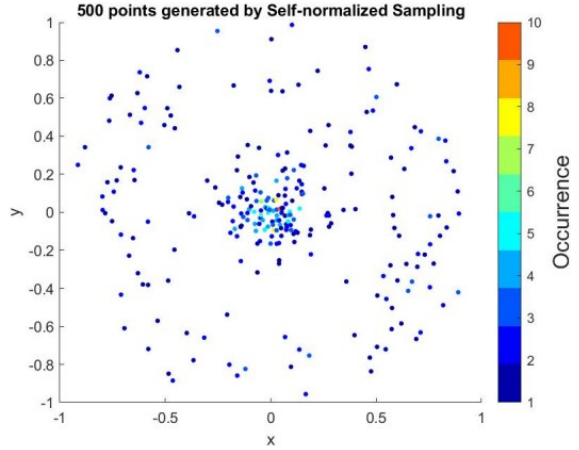
➢PINN still faces significant CoD issues in high-dimensions.

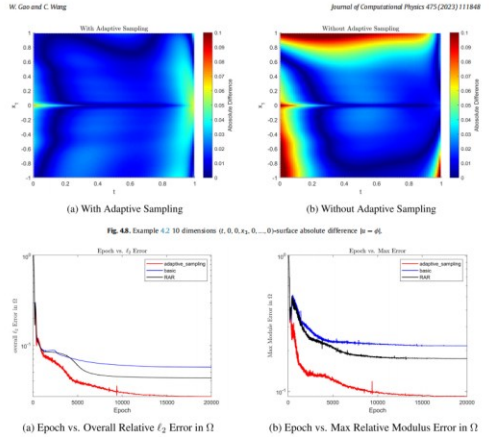
➢We need to sample a lot of collocation points (a lot of data).

➢Mitigate this by proposing a parallelizable self-normalized active learning algorithm

➢Efficient on GPUs.

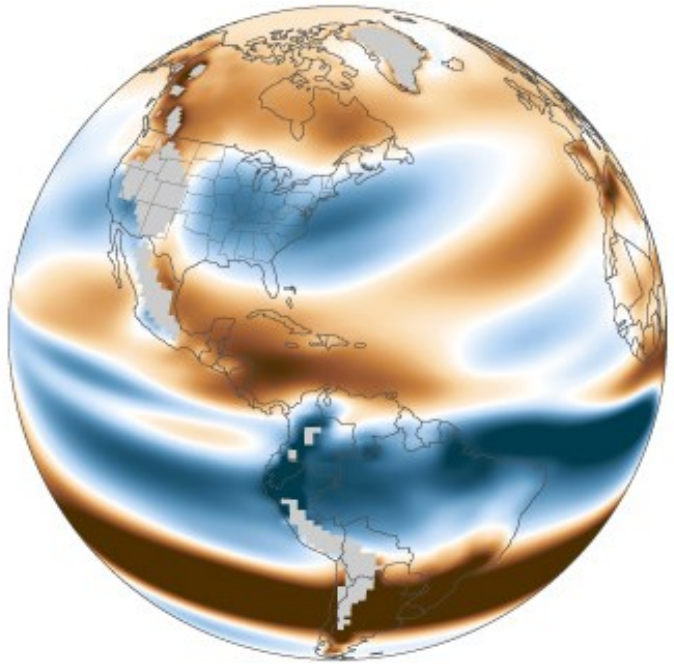






Neural Operators (Solving a Family of PDEs)

- Modeling physical systems under various parameters.
- A mapping from one function to another (data-driven!!!).
- Several orders of magnitude faster than numerical methods.



Example Usage in Climate Modeling:

Current atmospheric features → **Prediction for the next few days**

↓

Predicting natural disasters and saving lives!

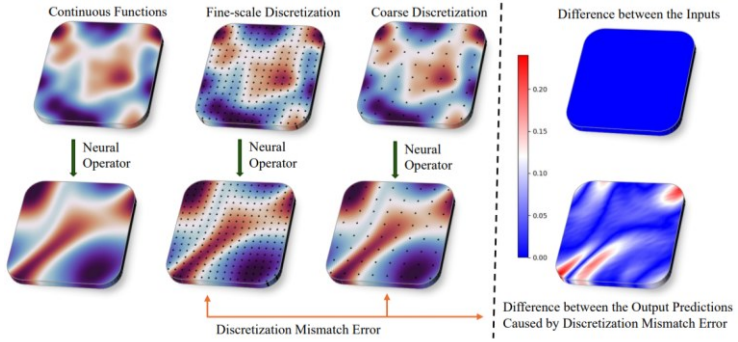
Stony Brook University

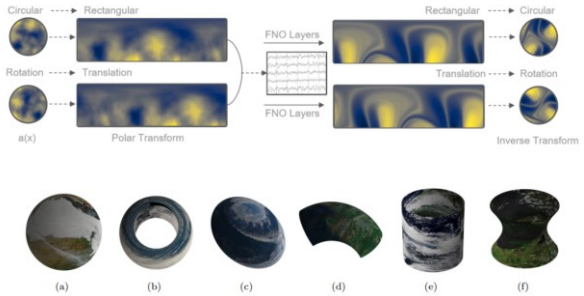
My Contribution

CROP, ICLR 2025 (Design):

Rebut the misunderstanding on discretization-invariance

Improved learning on multi-scale features





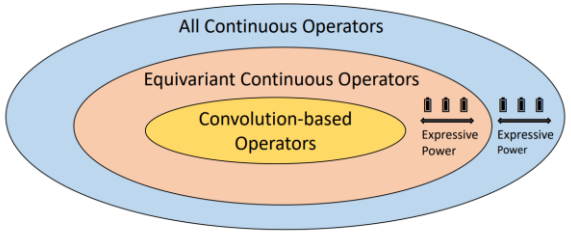
CT-FNO, TMLR 2024 (Design):

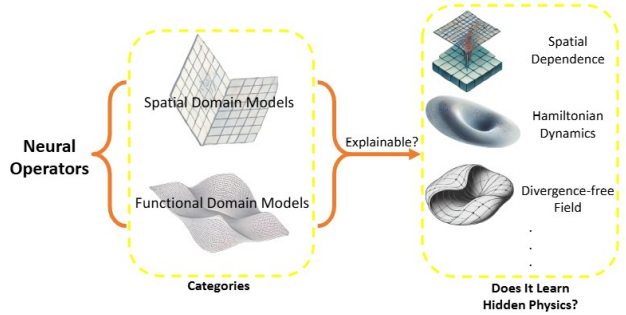
Incorporate symmetry priors into neural operators

Generalization to various irregular domains

DS-FNO, TMLR 2025 (Design):

Improved expressivity (learning capabilities) of neural operators





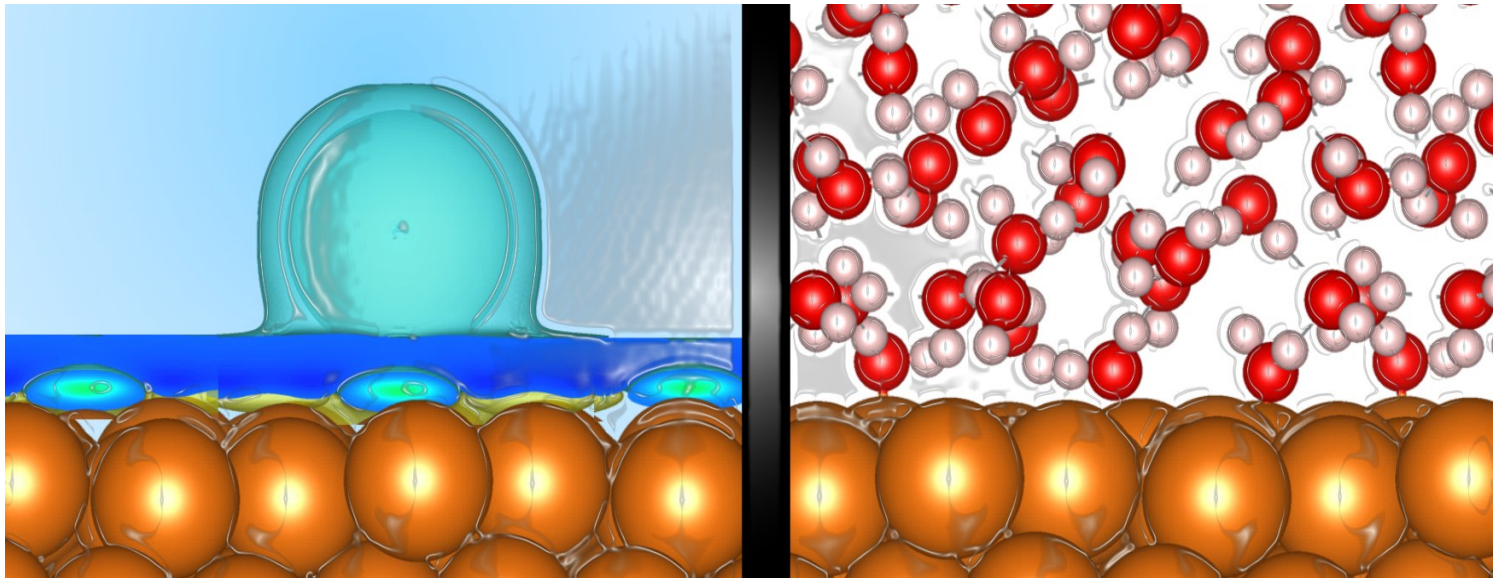
X-NO, Under Review (Trustworthiness):

Reveal if neural operators can learn the hidden dynamics in data

FAR BEYOND

Science at Different Scales

- The world is made of particles
- Macro-level → Continuum → PDEs to describe physical systems
  - What about micro-levels?



Continuum Atomistic

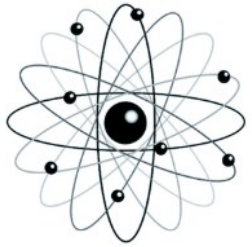
Talk Outline

- Introduction to AI and Neural Networks
- Introduction to AI4Sci and My Research
- Discriminative Models:
  - AI for Continuum Systems (PDEs)
    - Physics Informed Neural Networks
    - Neural Operators
  - AI for Atomistic Systems
- Generative Models
  - Introduction to Generative Models
  - Diffusion Models for Molecular Generation
  - LLMs for Science
- Acknowledgement

Representation of Atomistic Systems

Task Description:

Given an atomistic system, predict certain property values of this system.



Dipole moment

$\mu = 1.69D$

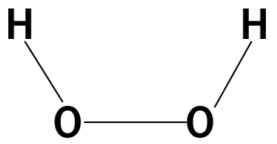
➢(Chemistry) The 3D geometric configuration is crucial in determining properties.

➢3D representations outperform their 2D counterparts by a large margin.

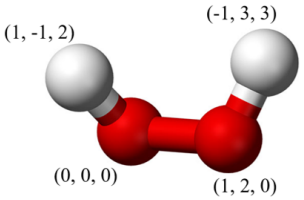
Representation of Atomistic Systems:

HOOH

H_2O_2 as 1D SMILE strings



H_2O_2 as 2D (planar) graphs

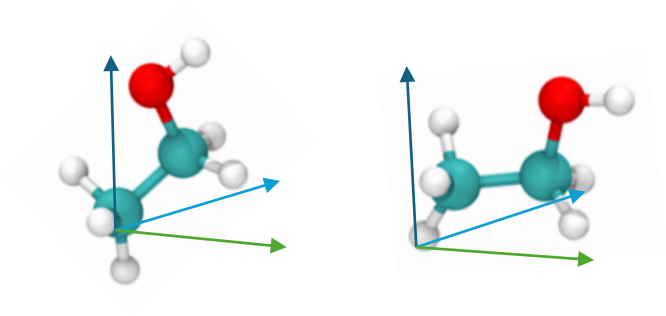


H_2O_2 as 3D (geometric) graphs

Model	MAE(Lower, better)	
GIN-Virtual	0.2371	→ Best 2D GNN
SchNet	0.1511	3D GNNs outperform 2D GNNs by a large margin
DimeNet++	0.1214	
SphereNet	0.1182	
ComENet	0.1273	

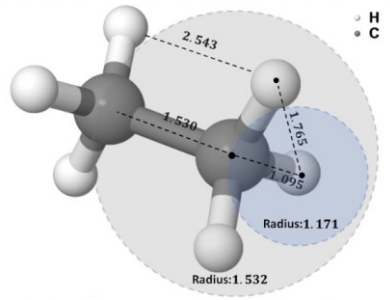
Learning with Geometric Representation

- Geometric representation → different designs of neural networks and different learning patterns
  - ➢ Symmetries: Roto-translation changes the coordinates, but does not change the molecule ← A lot of existing works on this already



→ Predictions can be completely different!

  - ➢Learning Patterns: Tensorial physical quantities (e.g. chemical forces, fields)



→ Learning chemical interaction patterns ← My Vision

Stony Brook University

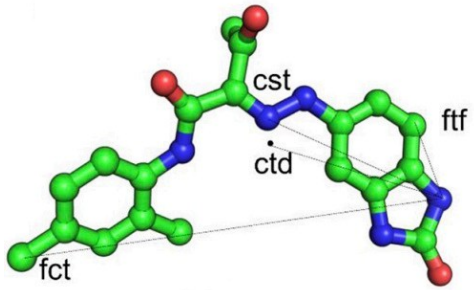
My Contribution

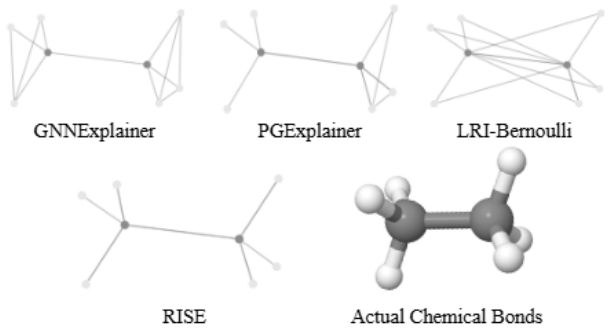
AL3DGraph, NeurIPS 2024 (Data):

Select the most informative training samples

Novel diversity and uncertainty metrics for 3D molecules

Reduce data by 50% while even improving performance





RISE, ICML 2025 (Trustworthiness):

Making AI models transparent to scientists

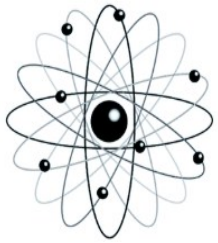
Reveal the learning patterns of geometric GNNs

Potential extension to discover unknown interactions

FAR BEYOND

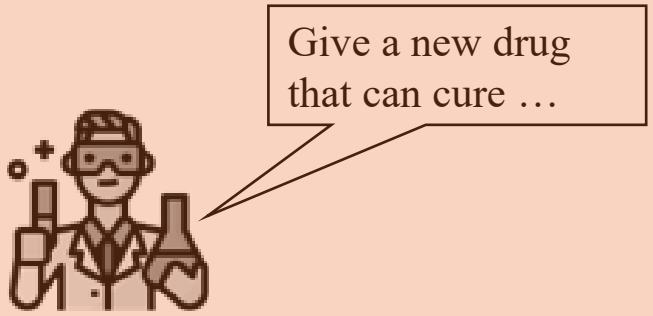
Heading Into the Era of Generative AI

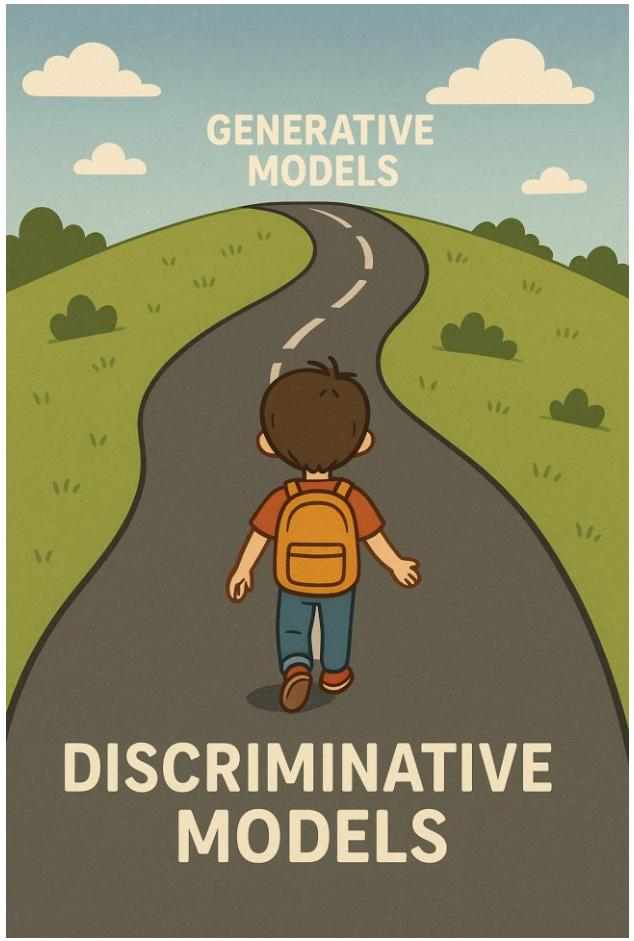
- Discriminative Models:
  Learn patterns from data and **select the best fit**



→ Dipole moment
$\mu = 1.69D$

- Generative Models:
  Learn patterns from data and **create new possibilities**





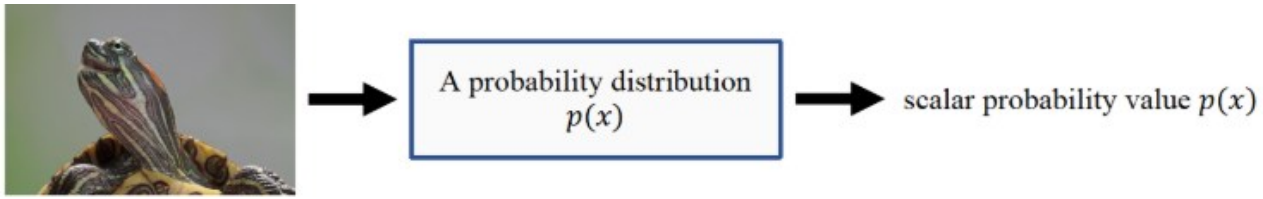
Talk Outline

- Introduction to AI and Neural Networks
- Introduction to AI4Sci and My Research
- Discriminative Models:
  - AI for Continuum Systems (PDEs)
    - Physics Informed Neural Networks
    - Neural Operators
  - AI for Atomistic Systems
- Generative Models
  - Introduction to Generative Models
  - Diffusion Models for Molecular Generation
  - LLMs for Science
- Acknowledgement

Introduction to Generative Models

Generative AI learns the underlying patterns of data to discover new possibilities beyond what already exists.

- (Statistical) Generative Models:
  - Learn distributions from data



- Generation by sampling

Introduction to Generative Models

🚫 Learning $p(x)$?

- **Extremely inefficient:** the target distribution is "peaky" and occupies a vanishingly small fraction of space.

Alternatives:

- Inverse Transform Sampling (e.g. VAE, Diffusion, Flow):
  - $x = f(z)$ where $z \sim p(z)$ is simple (e.g., Gaussian).
  - Data → Noise → Data
  - We approximate $f(z)$ with a neural network.
- Autoregressive Models (e.g. LLMs, PixelCNN)
  - $p(x) = \prod_i p(x_i \mid x_{<i})$
  - Generation by predicting one token/pixel/etc.. at a time.
  - The neural network learns similarly to a classification problem.

Talk Outline

- Introduction to AI and Neural Networks
- Introduction to AI4Sci and My Research
- Discriminative Models:
  - AI for Continuum Systems (PDEs)
    - Physics Informed Neural Networks
    - Neural Operators
  - AI for Atomistic Systems
- Generative Models
  - Introduction to Generative Models
  - Diffusion Models for Molecular Generation
  - LLMs for Science
- Acknowledgement

Introduction to Diffusion Models

- Diffusion models gradually add noise to data until it becomes pure Gaussian noise, then train a neural network to reverse the process by denoising step by step.
  - Generation: Pure Gaussian Noises → Realistic Data Samples

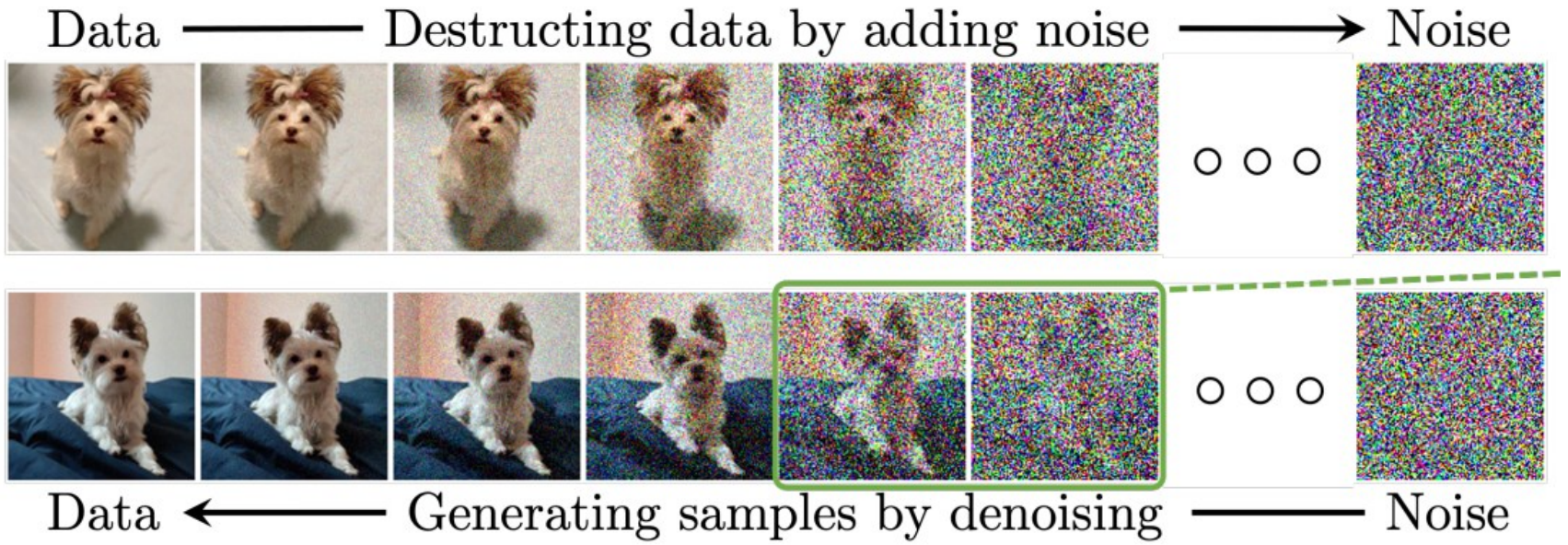
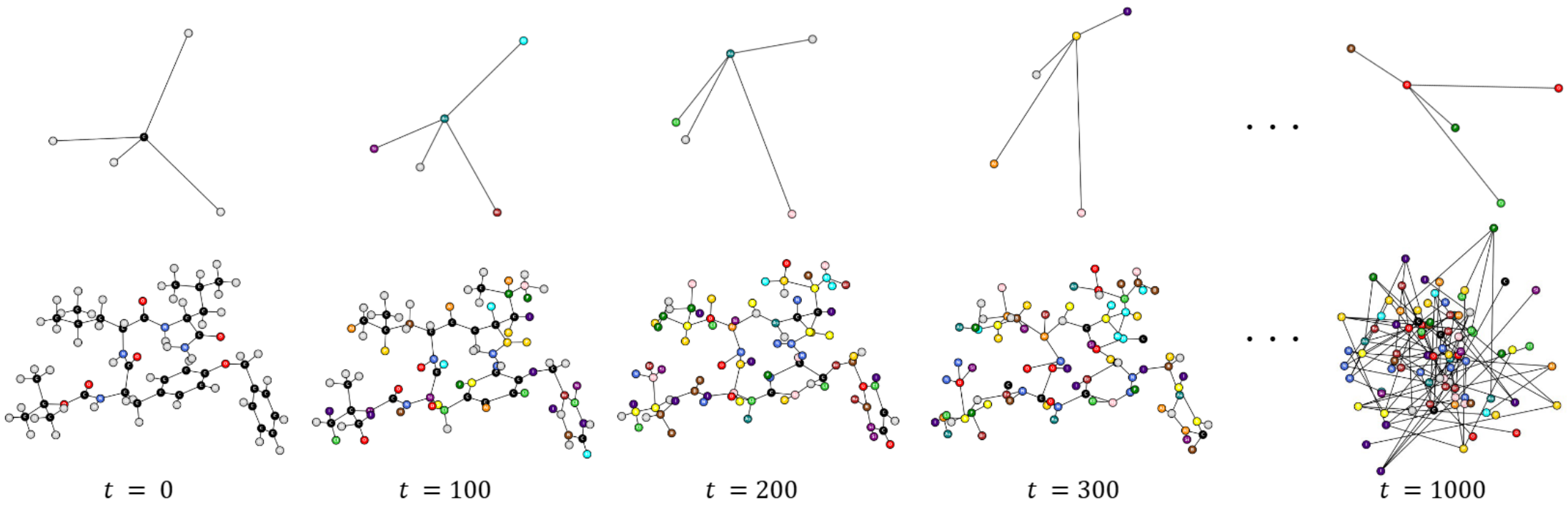


Image source: https://arxiv.org/pdf/2209.00796v13

3D Molecular Diffusion

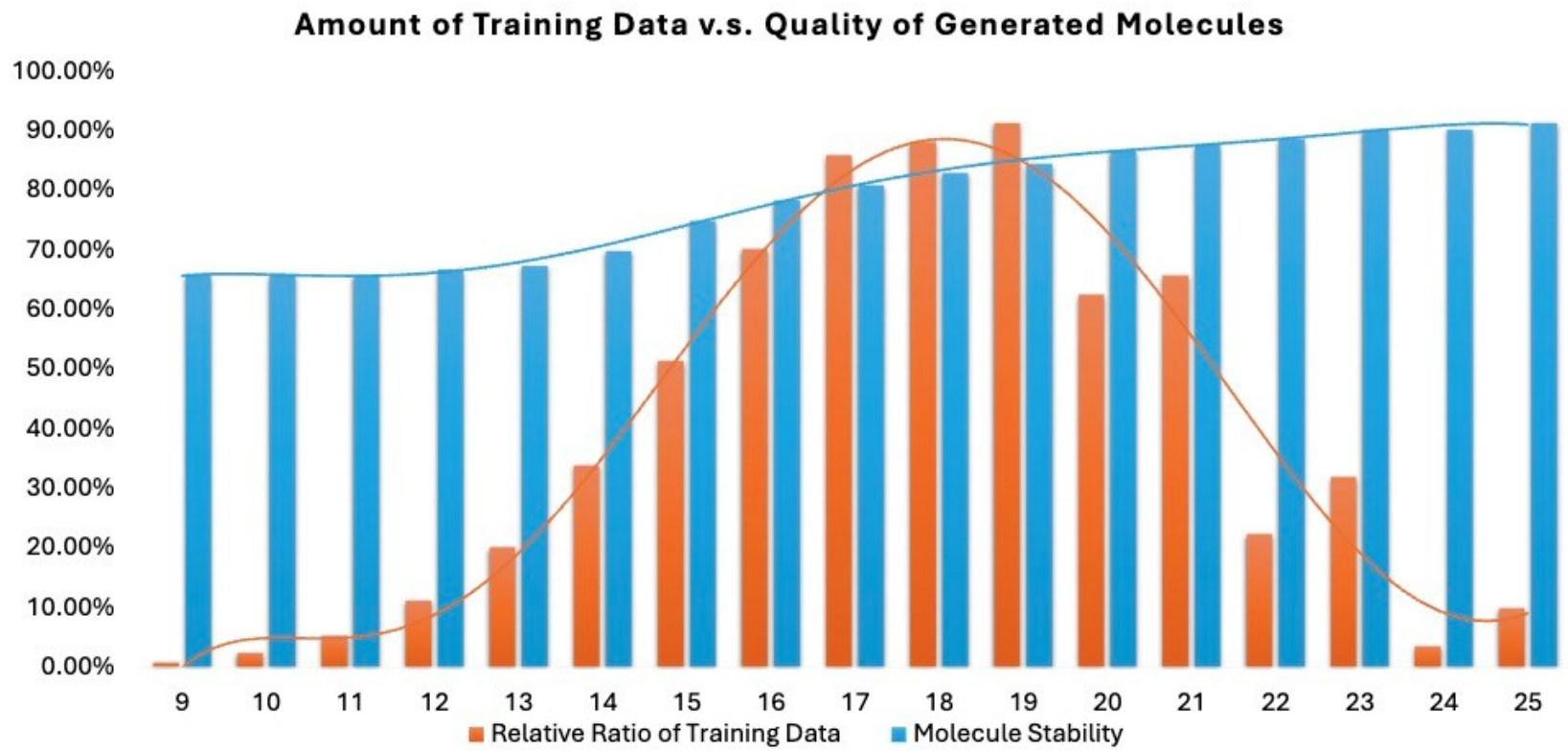
- Forward Process: Adding noise to gradually corrupt molecular structures.
- Reverse Process: Forming a coarse structure first and then fine-grained adjustments



Stony Brook University

Size-induced Inconsistency

- Larger molecules → Less data and higher structural complexity → Worse performance?

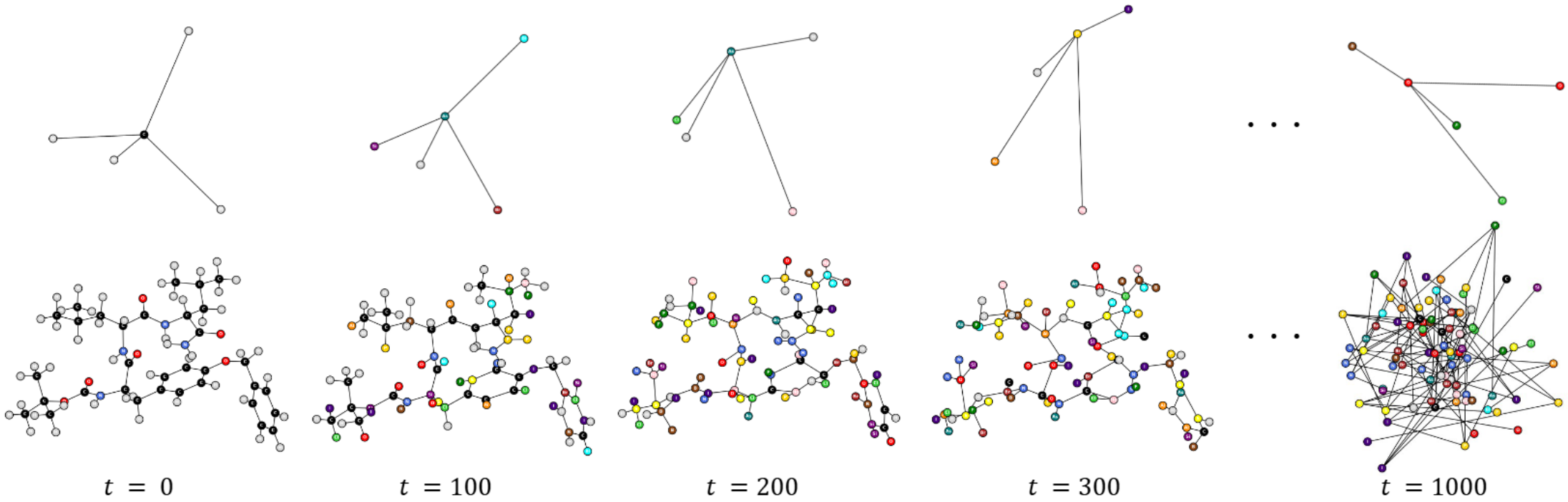


FAR BEYOND

Stony Brook University

Size-induced Inconsistency

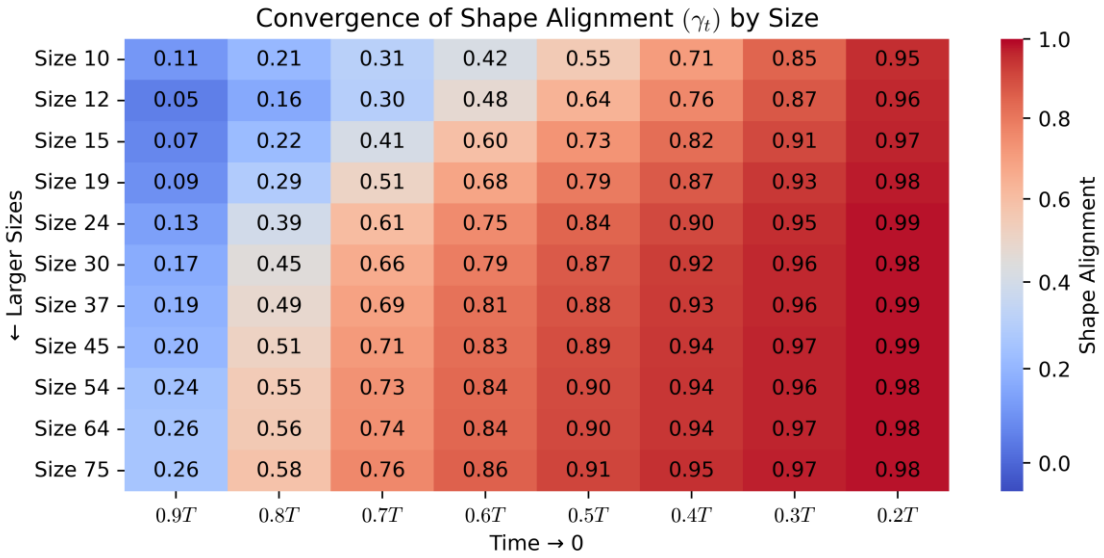
- Reverse Process: Forming a coarse structure first and then fine-grained adjustments

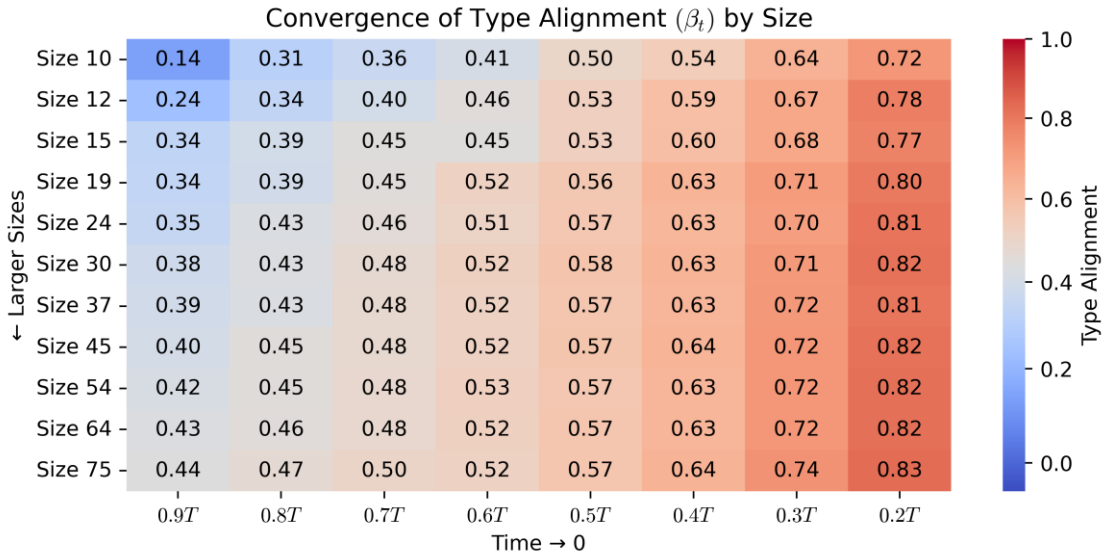


- The identity of the larger molecule is preserved longer/recovered earlier.

Size-induced Inconsistency

- Larger molecules stabilize earlier in the generative process.



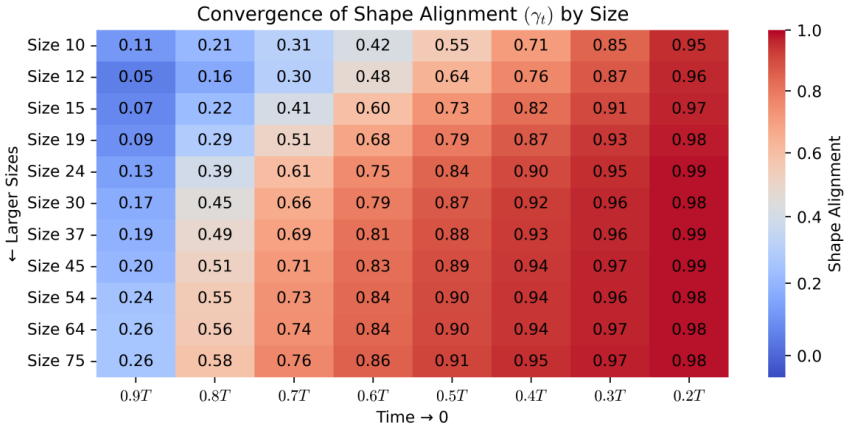


These numbers reflect how quickly the denoising trajectory “lines up” with the final molecule.

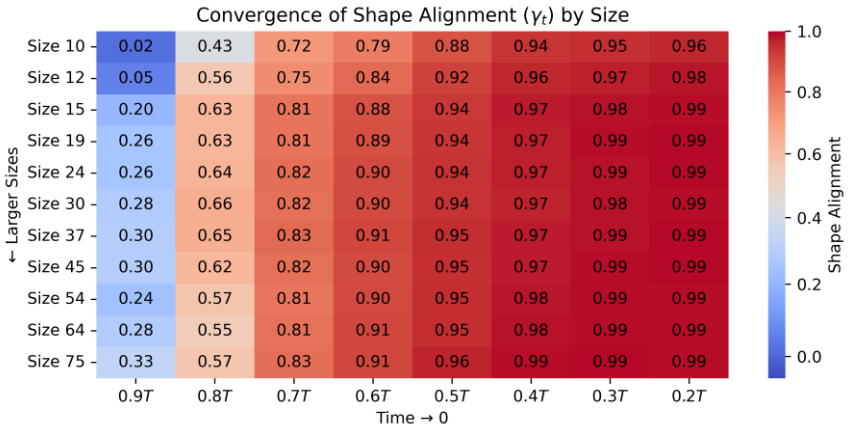
Scaling the Prior

➢We use Gaussian distributions of "smaller variance" for smaller molecules (Scaling the Prior).

Before:



After:



Improved generation quality for 3D molecular diffusion:

	QM9				GEOM-Drugs	
	Atom Stab (%)	Molecule Stab (%)	Valid (%)	Valid× Unique (%)	Atom Stab (%)	Valid (%)
Dataset	99.00	95.20	97.70	97.70	86.50	99.90
G-SchNet	95.70	68.10	85.50	80.30	-	-
ENF	85.00	84.90	40.20	39.40	-	-
EDM-bridge	98.80	84.60	92.00	90.70	82.40	92.80
EquiFM	98.90	88.30	94.70	93.50	84.10	98.90
GeoBFN	99.08	90.87	95.31	92.96	85.60	92.08
EDM	98.70	82.00	91.90	90.70	81.30	92.60
EDM-StP	98.83±0.03	88.07±0.22	94.41±0.08	92.63±0.14	84.11	95.59
RADM	98.50	87.30	94.10	91.70	85.00	99.30
RADM-StP	98.59±0.01	87.62±0.10	94.19±0.17	91.51±0.15	85.27	99.49
GeoLDM	98.90	89.40	93.80	92.70	84.40	99.30
GeoLDM-StP	99.08±0.05	90.70±0.22	95.41±0.16	93.49±0.16	86.78	99.37

Baseline results are taken from original works, some only have one decimal places available.

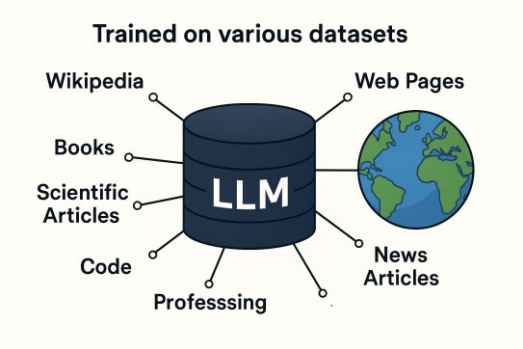
SOTA performance; improving simple diffusion models to even outperform complicated models

Talk Outline

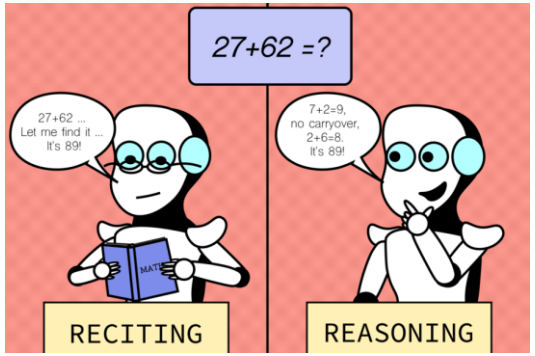
- Introduction to AI and Neural Networks
- Introduction to AI4Sci and My Research
- Discriminative Models:
  - AI for Continuum Systems (PDEs)
    - Physics Informed Neural Networks
    - Neural Operators
  - AI for Atomistic Systems
- Generative Models
  - Introduction to Generative Models
  - Diffusion Models for Molecular Generation
  - LLMs for Science
- Acknowledgement

LLMs for Science

- Why LLM-assisted Scientific Discoveries?

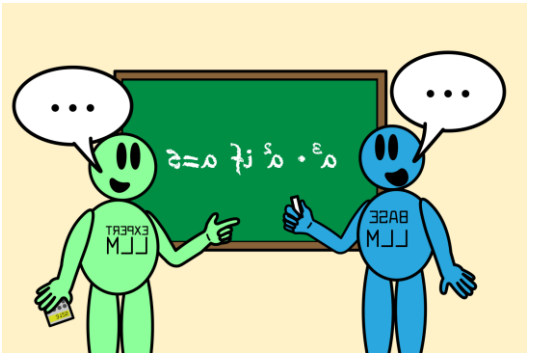


Absorbing Knowledge



Reasoning

Using Tools



Collaboration

- Challenges:
  - Lack of domain specialized knowledge
  - Difficulty handling multimodal inputs
  - Collaboration with domain tools or human experts

Stony Brook University

LLM-Assisted Drug Editing

Can you make the molecule O=C(NC[C@H]1CCCO1)c1ccccc1N1CCCC1=O more soluble in water? Please provide a short reasoning and give the result as a SMILES string.

To improve water solubility, we generally try to:

- **Increase polarity** by adding polar functional groups (e.g., hydroxyl, carboxyl, sulfonate).
- **Decrease hydrophobicity** by disrupting extended aromatic or aliphatic hydrophobic surfaces.

Your molecule has multiple aromatic and cyclic amide features that reduce solubility. A straightforward modification is to **add a hydroxyl group to the phenyl ring**, which increases polarity without drastically changing the structure.

Modified Molecule

I added an –OH group at the para position of the phenyl ring.

Resulting SMILES:

mathematica — Copy code

```
O=C(NC[C@H]1CCCO1)c1ccc(O)cc1N1CCCC1=O
```

```python
from rdkit import Chem
from rdkit.Chem import Crippen
# Define molecules
input_mol = Chem.MolFromSmiles("O=C(NC[C@H]1CCCO1)c1ccccc1N1CCCC1=O")
output_mol = Chem.MolFromSmiles("O=C(NC[C@H]1CCCO1)c1ccc(O)cc1N1CCCC1=O")

# Compute logP
input_logp = Crippen.MolLogP(input_mol)
output_logp = Crippen.MolLogP(output_mol)
if Chem.SanitizeMol(output_mol)==Chem.SanitizeFlags.SANITIZE_NONE:
  print("Output molecules is valid")
else:
  print("Output molecules is invalid")
print("Input logP:", input_logp)
print("Output logP:", output_logp)
print("ΔlogP (output - input):", output_logp - input_logp)
```

```
Output molecules is valid
Input logP: 1.7222
Output logP: 1.4278
ΔlogP (output - input): -0.2944
```

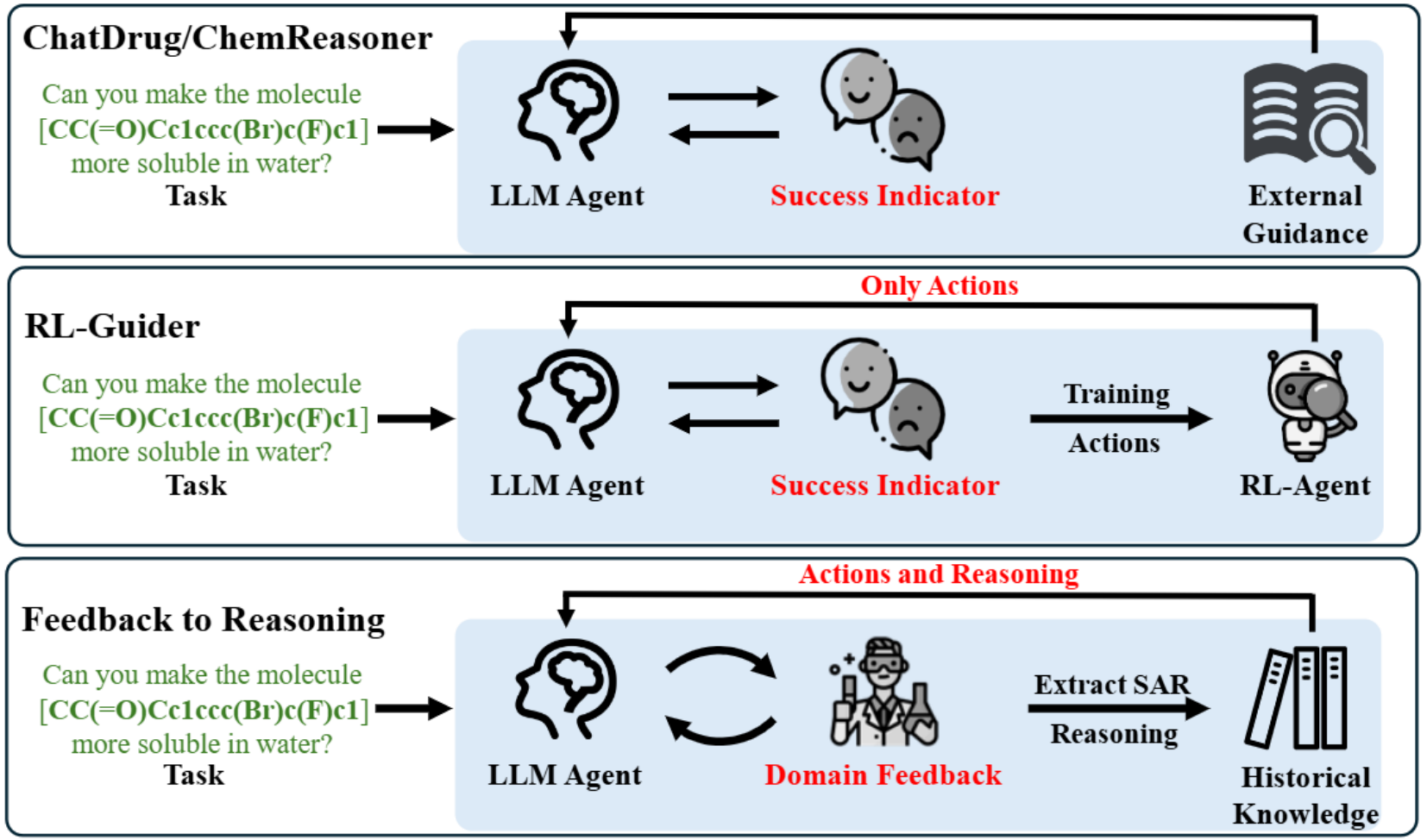
Lower logP → better water solubility

FAR BEYOND

Drug Editing

- Human Experts:
  - Try small changes to molecules
  - Get feedback from experiments or tests
  - Learn from mistakes and successes over time
  - Edit → Get Feedback → Learn and Self-reflect → Improve → Edit
- Existing LLM-Assistants:
  - Try small changes to molecules
  - Get informed whether the edit is successful
  - Edit → Flag Success → Edit

Our Pipeline: F2R



Our Pipeline: Feedback to Reasoning

- Lack of domain specialized knowledge
  - We provide detailed feedback to the LLM to inform its failures ← Improve in a single round
  - We incorporate a memory mechanism for LLM to learn from past experiences ← Improve across different sessions
- Difficulty handling multimodal inputs
  - Feedback is in text format through carefully designed templates
- Collaboration with domain tools
  - Autonomous collaboration with computational tools for the feedback without human intervention

Our Pipeline: Feedback to Reasoning

Task	Δ	ChatGPT-4.1					Gemini-2.5-Flash				
		Base LLM	Chat Drug	Chem Reasoner	RL-Guider	F2R	Base LLM	Chat Drug	Chem Reasoner	RL-Guider	F2R
More soluble in water	0	81.00	83.50	83.50	85.50	99.00	85.00	81.00	84.00	82.50	99.00
	0.5	84.00	81.50	84.00	83.50	96.00	80.50	81.50	76.50	79.50	96.00
Less soluble in water	0	85.00	85.50	84.50	85.50	99.00	95.50	97.00	98.00	91.50	99.00
	0.5	72.00	56.00	76.50	63.50	81.50	87.50	87.00	88.50	87.00	95.50
More like a drug	0	46.00	61.50	73.50	47.50	69.00	79.00	77.50	79.50	73.50	83.50
	0.5	6.00	20.00	18.00	8.50	21.00	16.50	27.00	22.50	19.50	30.50
Less like a drug	0	68.50	61.50	72.50	65.00	89.00	70.50	68.50	85.50	69.50	78.50
	0.1	16.50	28.50	52.00	24.50	63.50	44.00	43.00	67.00	53.50	65.00
Higher permeability	0	31.50	53.50	81.50	47.50	94.50	92.50	91.00	91.00	93.00	97.00
	10	19.50	36.50	62.50	34.00	74.00	52.50	62.00	63.00	61.50	79.00
Lower permeability	0	87.00	85.50	88.00	86.50	99.00	86.00	86.50	83.50	84.50	99.00
	10	87.00	83.50	88.50	86.50	97.50	85.00	81.50	82.00	84.50	98.50
More hydro-bond acceptors	0	74.00	69.00	76.50	77.50	97.00	80.50	82.50	78.50	74.50	99.00
	1	19.00	23.00	34.00	20.50	42.50	44.00	44.00	57.00	44.50	68.50
More hydro-bond donors	0	80.00	78.00	85.50	81.00	97.50	74.50	70.50	75.00	70.50	98.00
	1	13.00	26.50	19.50	22.50	41.50	16.50	15.00	47.00	15.50	52.50

Improve the performance significantly
Achieve even 99% success rates for some tasks

Talk Outline

- Introduction to AI and Neural Networks
- Introduction to AI4Sci and My Research
- Discriminative Models:
  - AI for Continuum Systems (PDEs)
    - Physics Informed Neural Networks
    - Neural Operators
  - AI for Atomistic Systems
- Generative Models
  - Introduction to Generative Models
  - Diffusion Models for Molecular Generation
  - LLMs for Science
- Acknowledgement

Stony Brook University

Acknowledgement

Ph.D. Advisor

Dr. Yi Liu
Professor of Data Science

Close Research Collaborators

Jingxiang Qu
Ph.D. Student, CS

Fang Wan
Ph.D. Student, CS

Xufeng Liu
Ph.D. Student, CS

Ruichen Xu
Ph.D. Student, AMS

FAR BEYOND