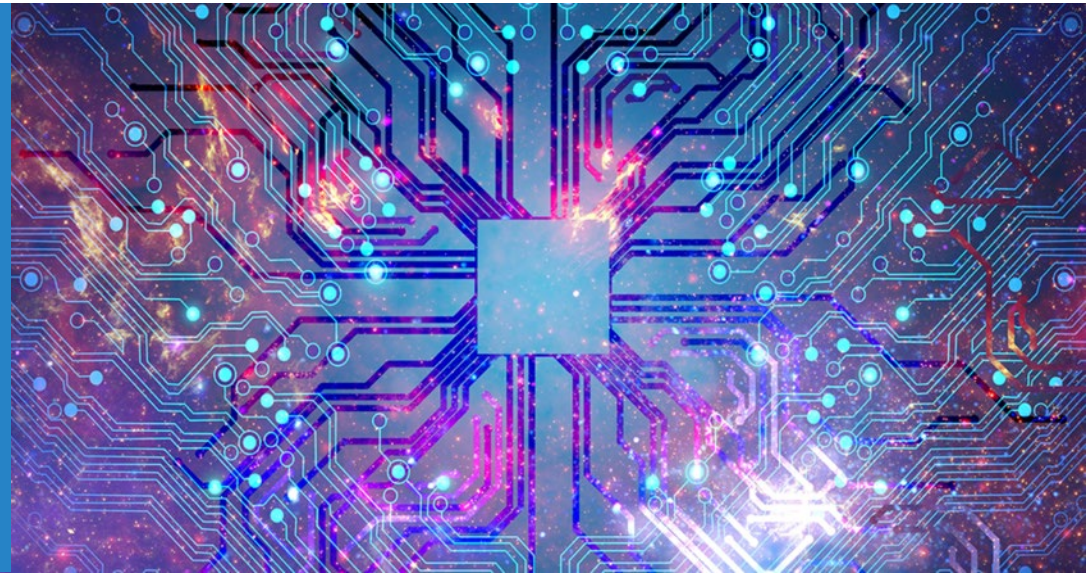
Biological Data in the Era of Foundation Models, Embodied AI and Autonomous Laboratories



Arvind Ramanathan/ ramanathana@anl.gov

Argonne National Laboratory/ University of Chicago Consortium for Advanced Science and Engineering (CASE)

Northwestern-Argonne Institute for Science and Engineering (NAISE)

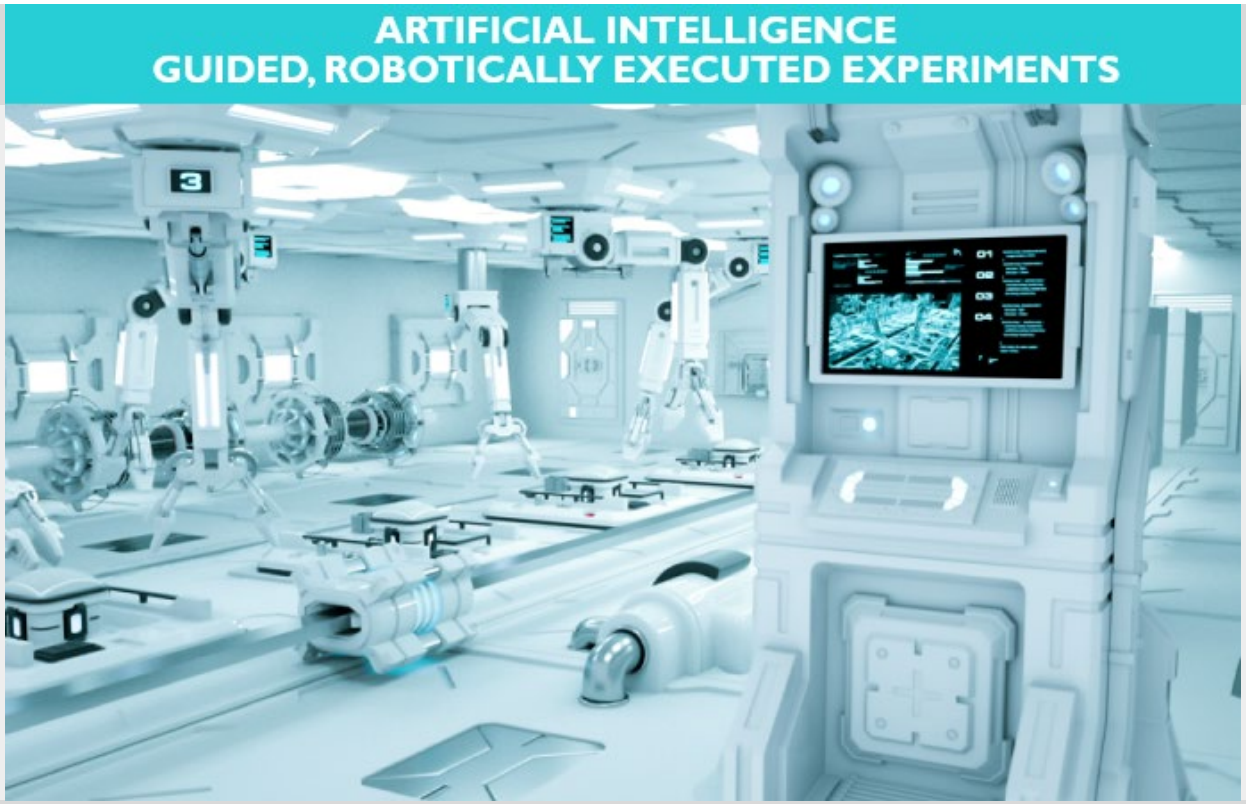
The Traditional vs. Emerging Paradigms of Scientific Discovery

ARTIFICIAL INTELLIGENCE
GUIDED, ROBOTICALLY EXECUTED EXPERIMENTS

- The current paradigm for discovery is inefficient, time-consuming, labor intensive, and costly
- The level of specialization required to explore a large hypothesis space limits researchers
- Data sets are clustered, sparse, less-interpretable, and incomplete
- Reproducibility crisis

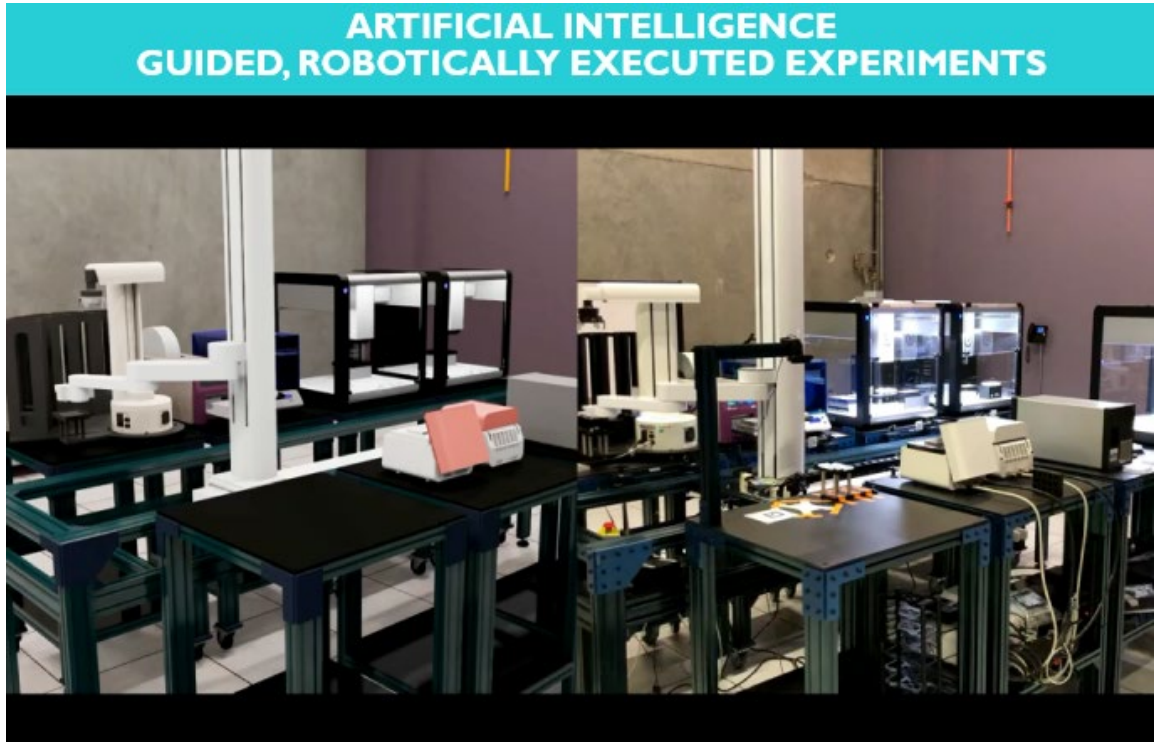
The Traditional vs. Emerging Paradigms of Scientific Discovery

- Accelerate the discovery process
- Elevate human creativity to higher level goals
- Democratize biological/materials synthesis and characterization
- Unbiased data collection and evaluation



Autonomous Discovery @Argonne

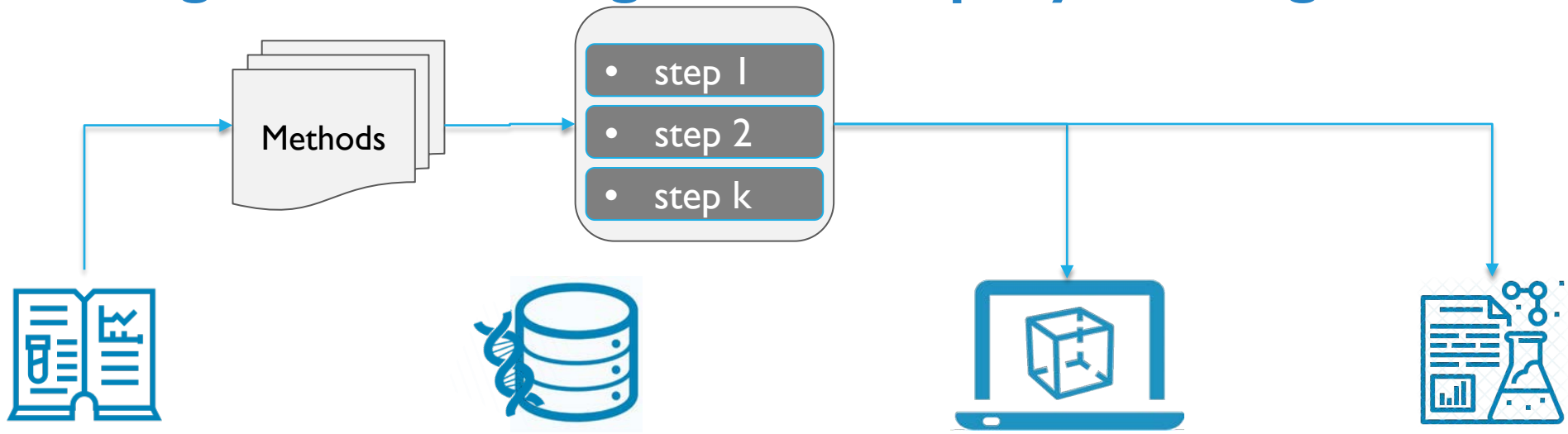
- **The vision**
  - A system that starts with a high-level description of a hypothesis and autonomously carries out computational and experimental workflows to confirm or reject that hypothesis
  - **Use of AI in robotics and simulations to close the loop on planning, execution, and analysis of experiments**
- Builds on
  - **AI approaches to planning** (multiple steps), and integration of results, causality, etc.
  - **Machine learning/simulation** to design and predict exp properties and outcomes
  - **Automation of experimental protocols** (robotic steps and workflows)
  - **Active Learning or RL** for selection of next experimental targets, etc.



https://github.com/anl-sdl/
https://www.cs.uchicago.edu/~rorymb/

Outline

How biological data management is rapidly evolving?



Scientific literature

Databases, Knowledge base/ontologies, data stores

Simulations

High throughput/ automated labs/ facilities

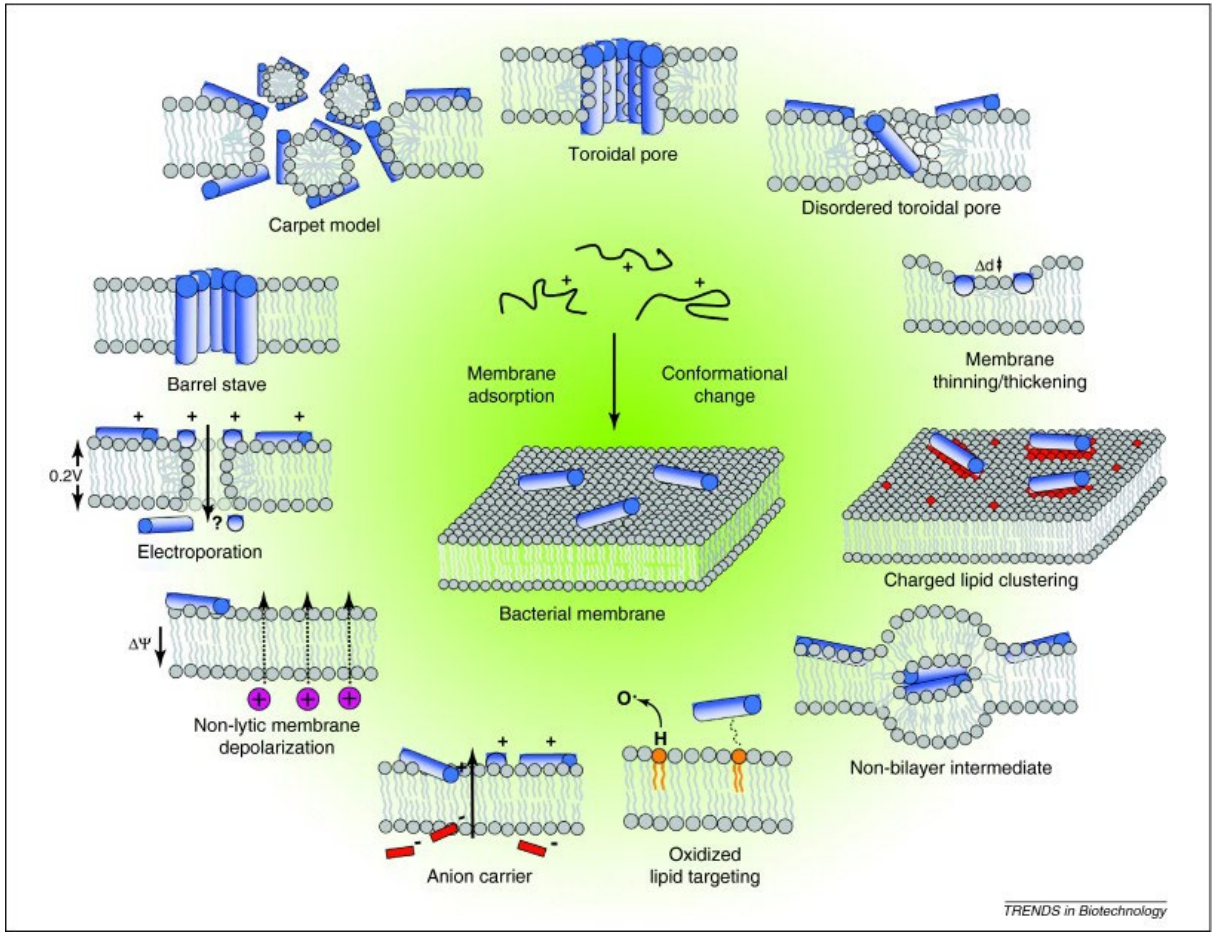
Design of antimicrobial peptides

An antimicrobial peptide (AMP) is a short (typically 12 to 50 amino acid) molecule that can target and kill viruses, bacteria, fungi, and other pathogens

Challenge: Design an AMP that can kill specified bacterial strains without harming host cells

With 20 possible amino acids, there are $20^{20} = 10^{26}$ AMPs of length 20

A rational design approach might combine knowledge of bacterial cell membrane composition and structure, AMP molecular and structural properties, host cell membrane characteristics and intracellular pathways—knowledge that may be gained by database/literature search, simulation, experiment



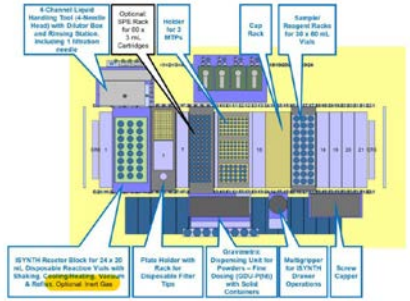
L. T. Nguyen, E.F. Haney, H.J Vogel, The expanding scope of antimicrobial structures and their modes of action, Trends in Biotechnology, 20 (9): 464-472

Automated synthesis and screening platform for antimicrobial peptides design

1

2

Throughput: ~96 peptides/day



Amide Coupling Screening – ISYNTH REACTSCREEN (SWING based)

Peptide synthesis (**PSE**)

3

Throughput: 7 strains x 96 peptides/ day + (16-24 hours)

Plate incubator

Hudson robotic arm set up for liquid handling

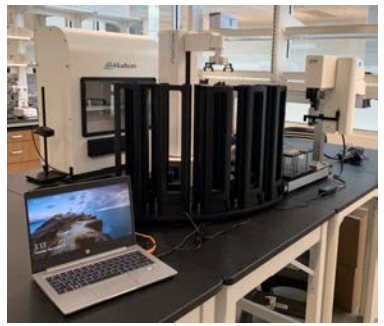
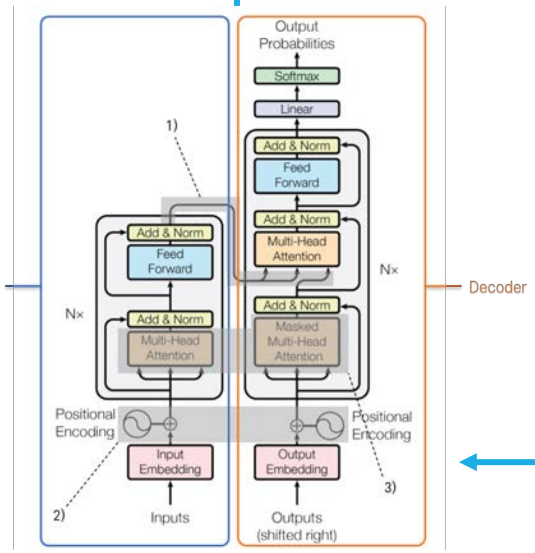


Plate reader

Peptide screening across ~30 Hope College E. coli strains + NIH/CDC ESKAPE collection (**BIO**)

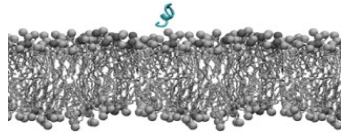
AI models learn successful features from current experiments and constrain generation of AMPs every cycle



1A

Large-language models for AMP generation (**ALCF**)

Throughput: O(1000s) per hour



Simulations capture behaviors of successful (and unsuccessful) experiments and derive features for constraining AI models

Simulations + simulation surrogates (**ALCF**) *Throughput: O(10s) per day*

Growth assay application

Creator

Dates

- Apr 05 2023 1
- Apr 06 2023 1
- Apr 07 2023 1
- May 15 2023 2
- May 30 2023 2
- May 31 2023 4
- Jun 01 2023 1
- Jun 02 2023 1
- Jun 13 2023 4
- Jul 18 2023 21
- Jul 19 2023 5
- Jul 20 2023 7
- Jul 21 2023 5

T0_Reading_1_16_49_31

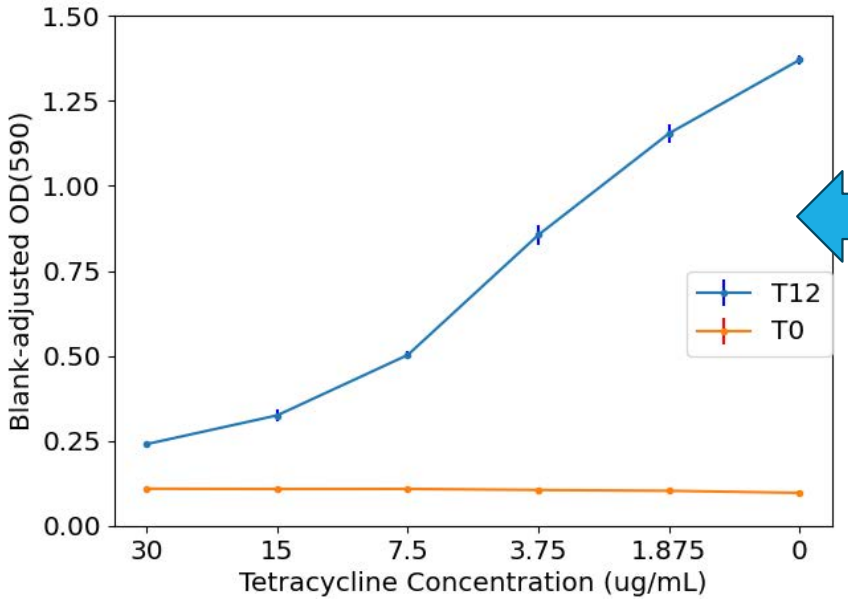
T0_Reading_2_15_11_52

T0_Reading_1_14_33_37

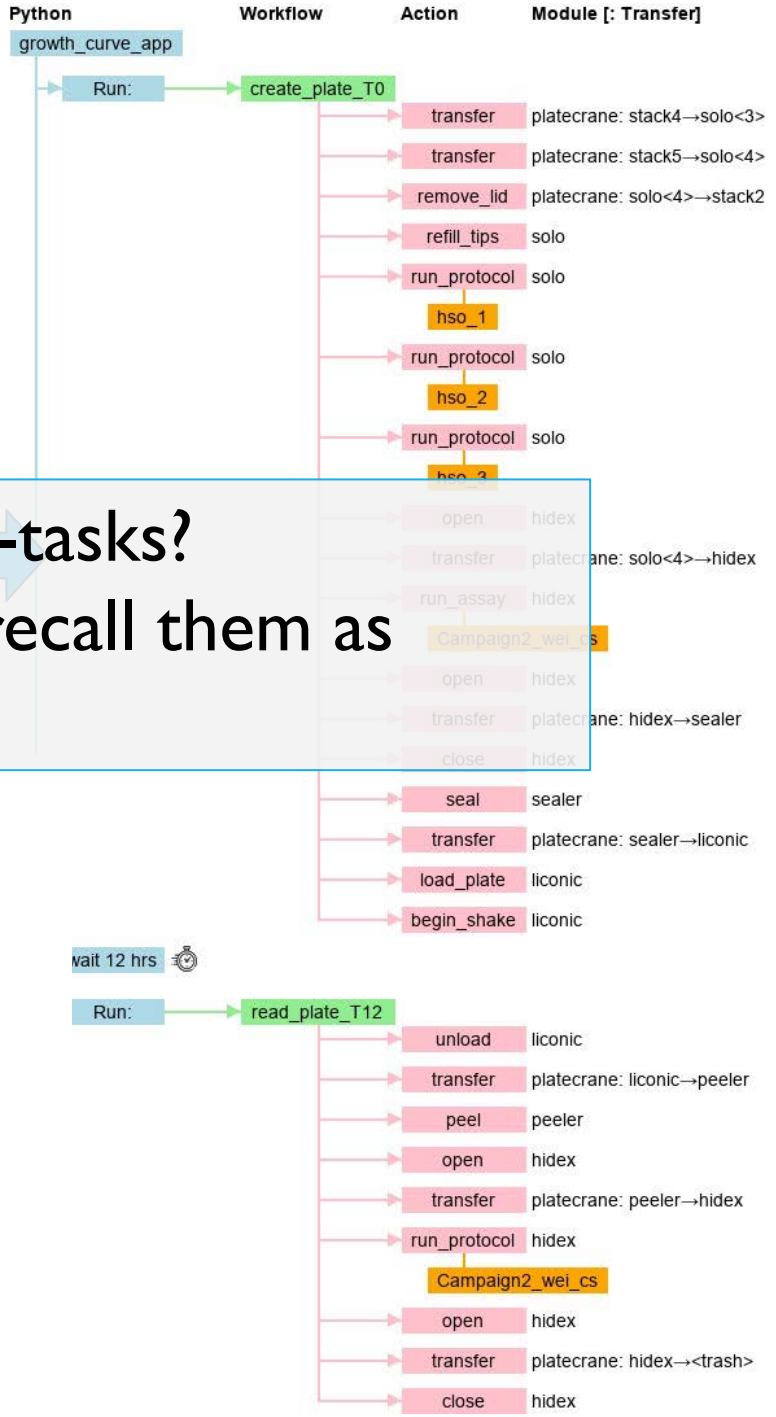
T0_Reading_2_18_01_28

List of datasets, one per experiment, on data portal.

1. Can we translate lab protocols into a list of lab sub-tasks?
2. Given list of sub-tasks, can we solve each task and recall them as skills?



Results from experiment in which tetracycline solution at varying concentrations was added to E. coli. Y-axis = blank-adjusted optical density at 590nm at start of experiment (T0) and 12 hours after start (T12). Results show mean plus error bars from four identical runs.



Embodied Agent for Automated Lab Code Generation

```
#################### Performing Task 1... ####################
Reasoning: Based on the information provided, it seems like the next logical step
 would be to prepare the master mix for the PCR reaction. This involves combining
 various reagents in specific volumes to create the master mix solution.

Task: Prepare the master mix for the PCR reaction.
```

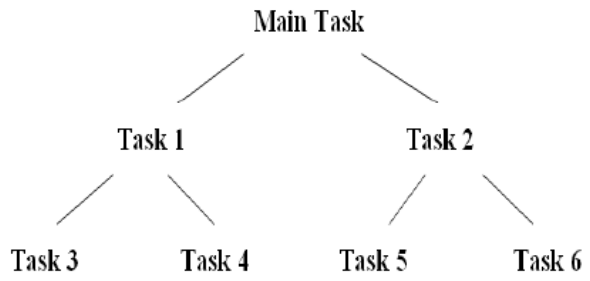
Agent

Memory

Task Prompt

Task Decomp.

Goal Tracking

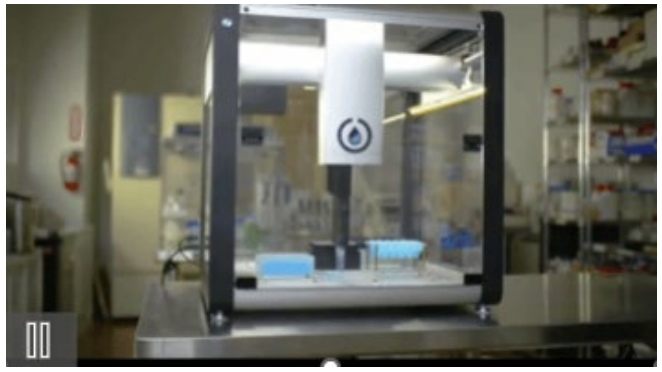


Candidate Code

```
Useful Programs:

def PCR_Master_Mix(labware_info, protocolContext):
    */
    Input: labware_info --> json-str
    Pass in a variable labware_info that contains labware information and
    quantities used

    Output: function call that creates master mix DNA and assigns to appropriate locations
    */

Human:
labware_info = {"number_of_samples":96,
"right_pipette":"flex_8channel_1000",
"left_pipette":"flex_8channel_1000","mastermix_volume":18,"DNA_volume":2}
```

Code Action

Execution Error

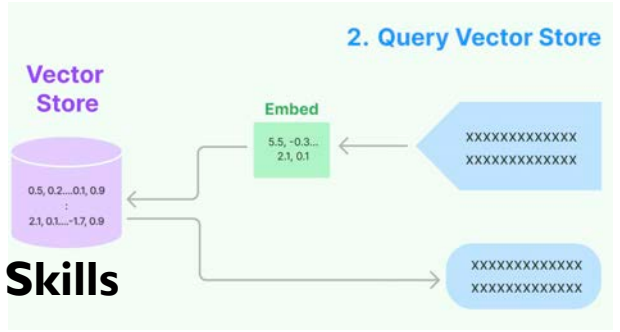


Verify Code

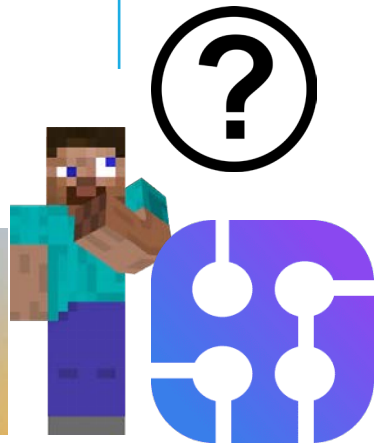
Refine Code

Retrieve

Memory of Tasks

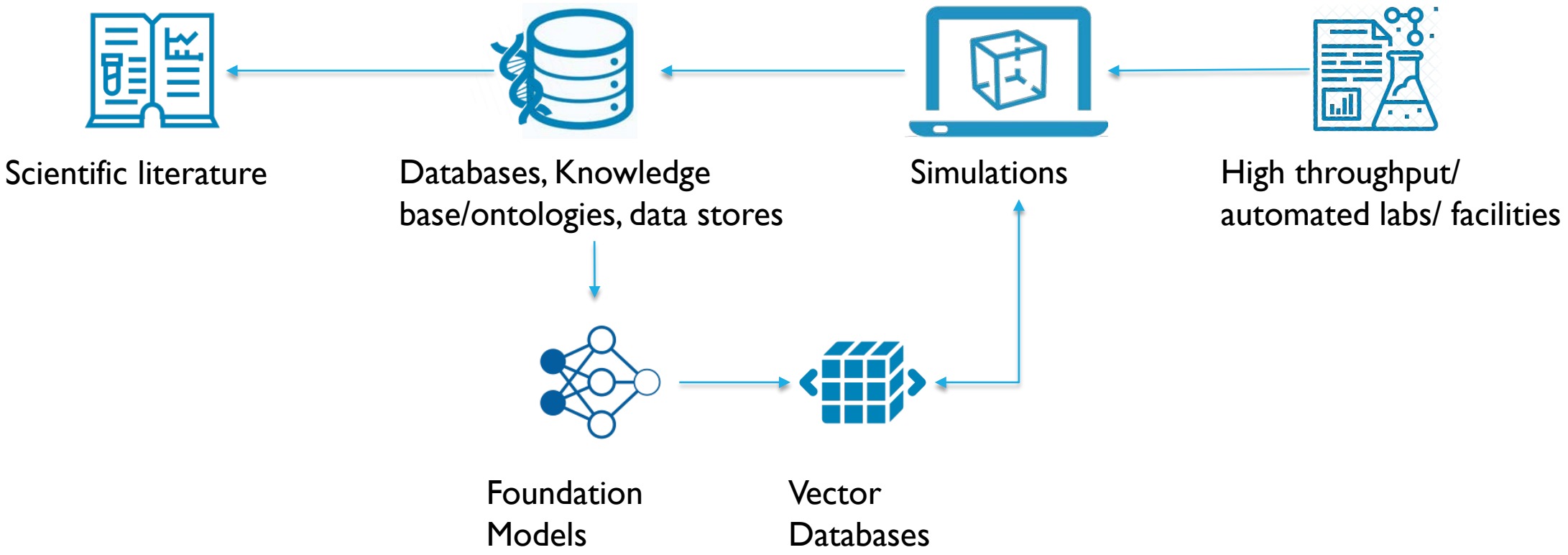


Add Code Skill

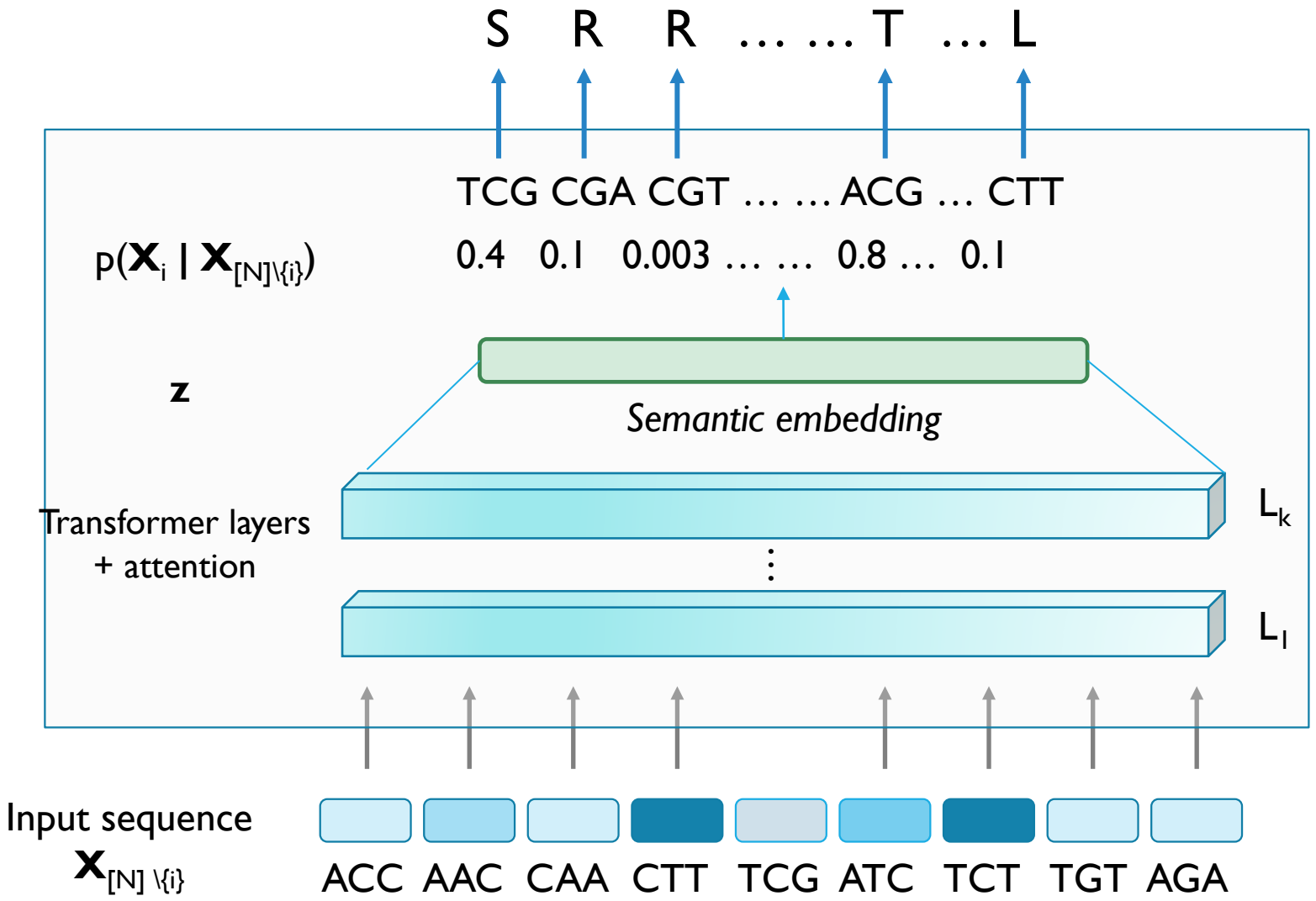


Outline

How biological data management is rapidly evolving?



Genome-scale Language Models (GenSLMs)



Model	Seq. length	#Parameters	Dataset
GenSLM-Foundation	2048	25M, 250M, 2.5B, **25B**	110M
GenSLM	10240	25M, 250M, 2.5B, **25B**	1.5M
GenSLM-Diffusion	10240	2.5B	1.5M

- Scaling LLMs with 25B parameters:
  - O (L^2) complexity in the attention computation
  - overcome communication overheads, parameters, checkpointing
- Variation within SARS-CoV-2 sequences can be small (< 1% overall variation)
  - Need foundation model to accommodate diversity
- **One of the largest foundation model trained on raw nucleotide sequences**

Infrastructure of GenSLM Foundation Models

Transformers

build passing | license Apache-2.0 | website online | release v2.0.0

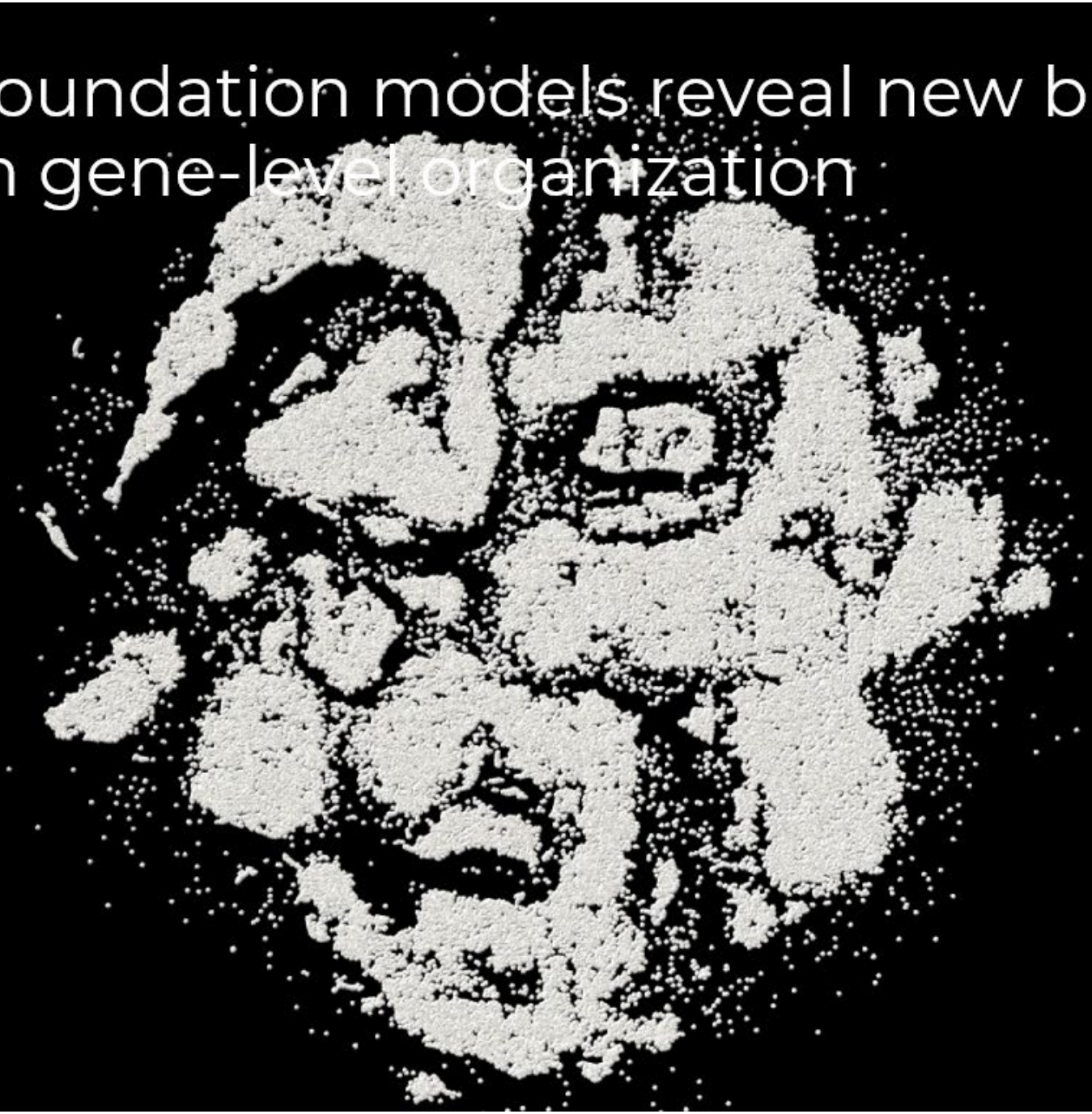
Lightning

```python
import torch
import numpy as np
from torch.utils.data import DataLoader
from genslm import GenSLM, SequenceDataset

model = GenSLM("genslm_25M_patric", model_cache_dir="/content/gdrive/MyDrive")
model.eval()

# Input data is a list of gene sequences
sequences = [
    "ATGAAAGTAACCGTTGTTGGAGCAGGTGCAGTTGGTGCAAGTTGCGCAGAATATATTGCA",
    "ATTAAAGATTTCGCATCTGAAGTTGTTTTGTTAGACATTAAAGAAGGTTATGCCGAAGGT",
]

dataset = SequenceDataset(sequences, model.seq_length, model.tokenizer)
dataloader = DataLoader(dataset)

# Compute averaged-embeddings for each input sequence
embeddings = []
with torch.no_grad():
    for batch in dataloader:
        outputs = model(batch["input_ids"], batch["attention_mask"], output_hidden_states=True)
        # outputs.hidden_states shape: (layers, batch_size, sequence_length, hidden_size)
        emb = outputs.hidden_states[0].detach().cpu().numpy()
        # Compute average over sequence length
        emb = np.mean(emb, axis=1)
        embeddings.append(emb)

# Concatenate embeddings into an array of shape (num_sequences, hidden_size)
embeddings = np.concatenate(embeddings)
embeddings.shape
>>> (2, 512)
```

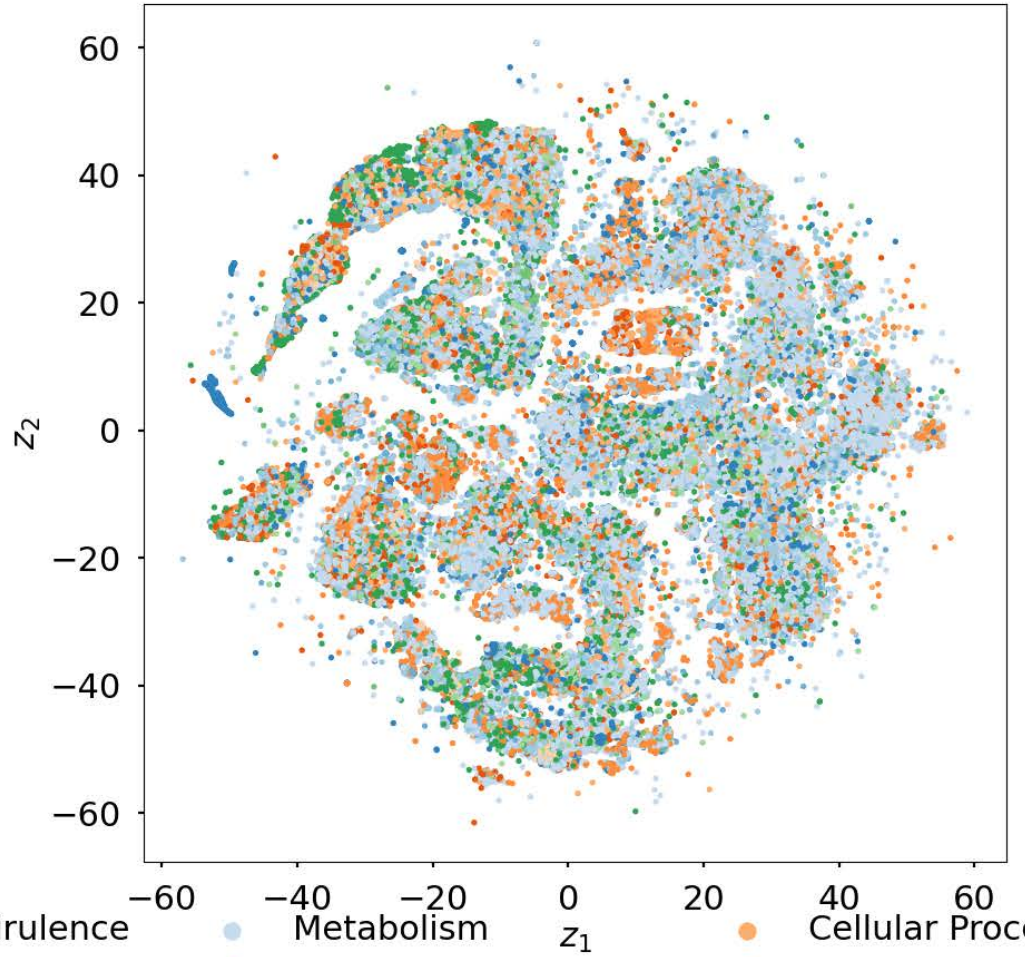
deepspeed

1

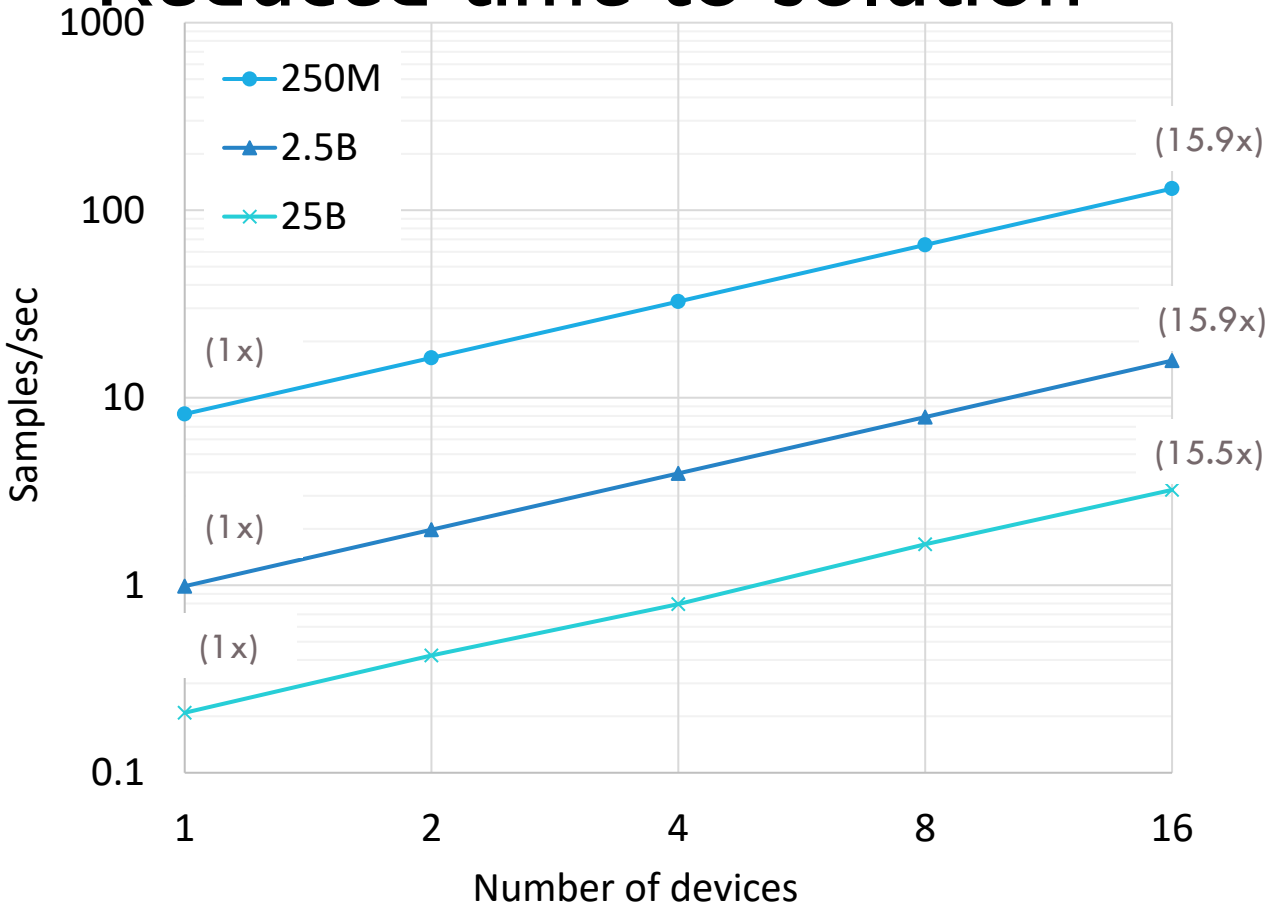
GenSLM Foundation models reveal new biological insights on gene-level organization



GenSLMs also reveal function level organization of genes

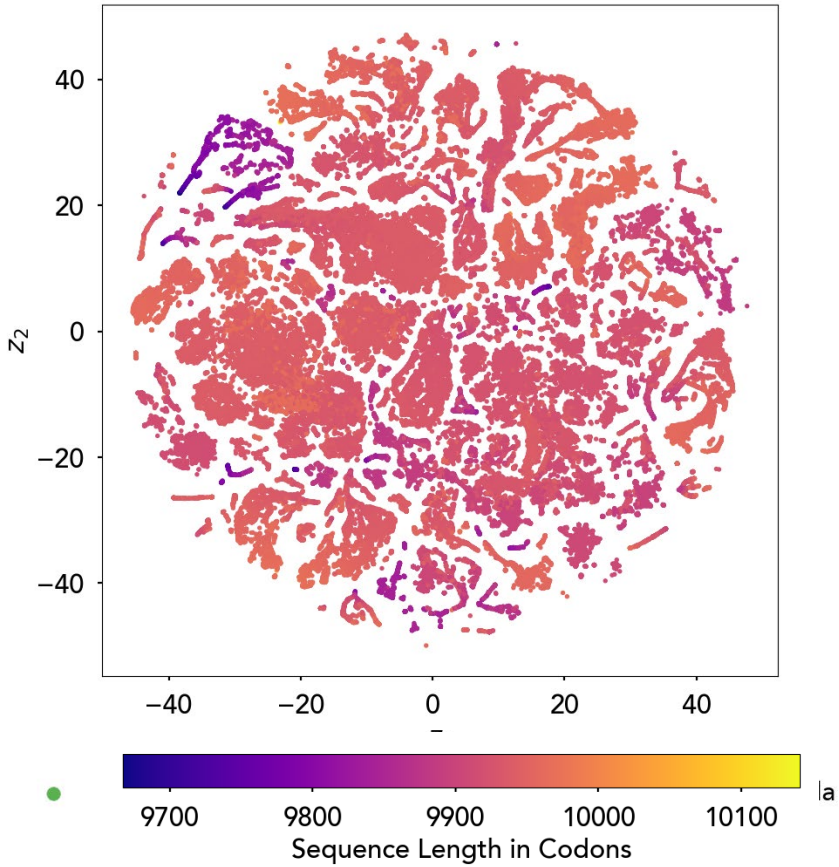


Reduced time to solution



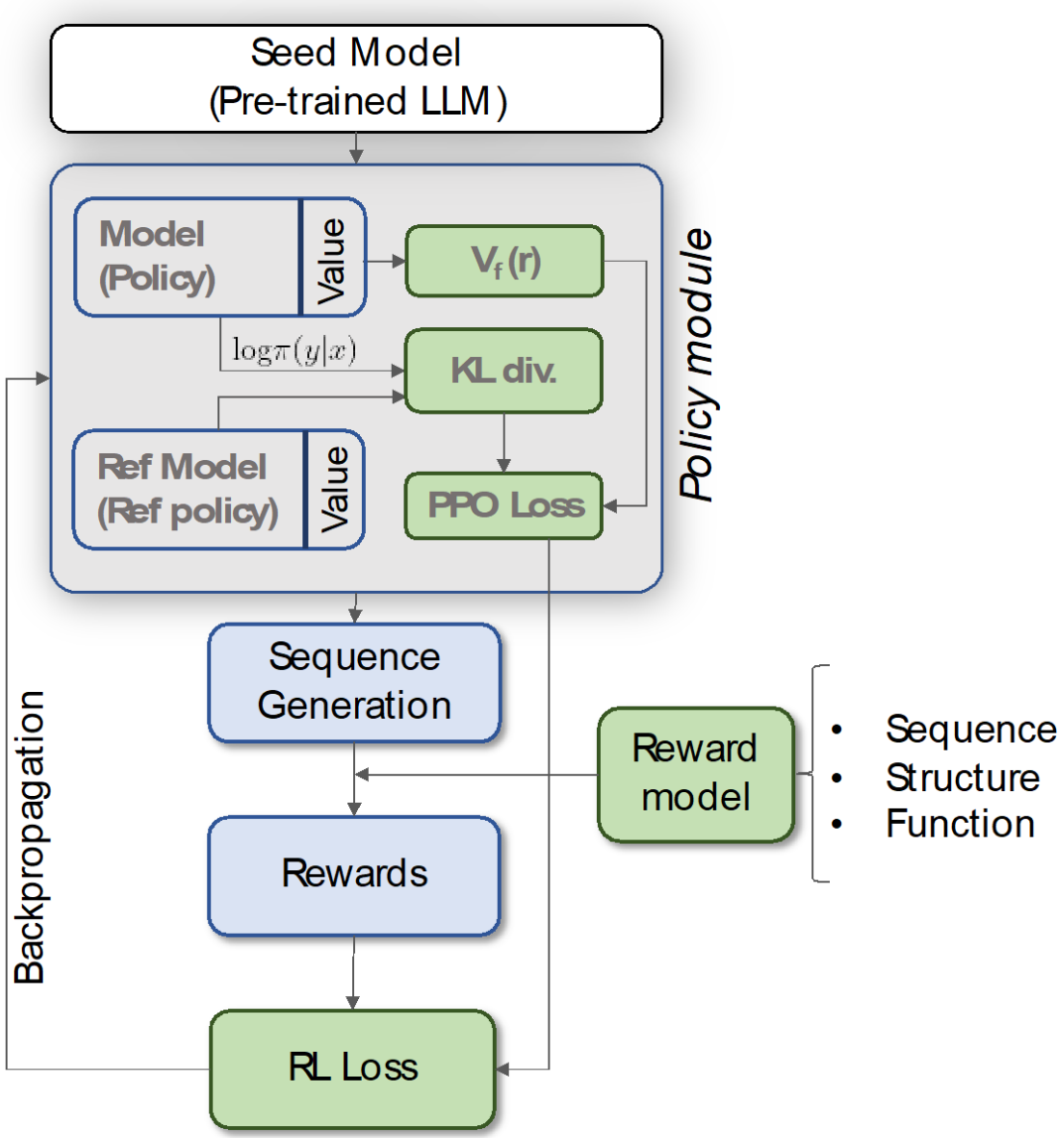
Fast time-to-solution: Train from scratch with MSL 10,240 in less than half a day with 4 CS-2s

Embeddings produced by GenSLM 1.3B, MSL 10,240



	GenSLM 123M		GenSLM 1.3B	
	1 CS-2	4 CS-2	1 CS-2	4CS-2
Training steps	5,000	3,000	4,500	3,000
Training samples	165,000	396,000	49,500	132,000
Time to train (h)	**4.1**	**2.4**	**15.6**	**10.4**
Validation accuracy	0.9615	0.9625	0.9622	0.9947
Validation perplexity	1.031	1.029	1.031	1.025

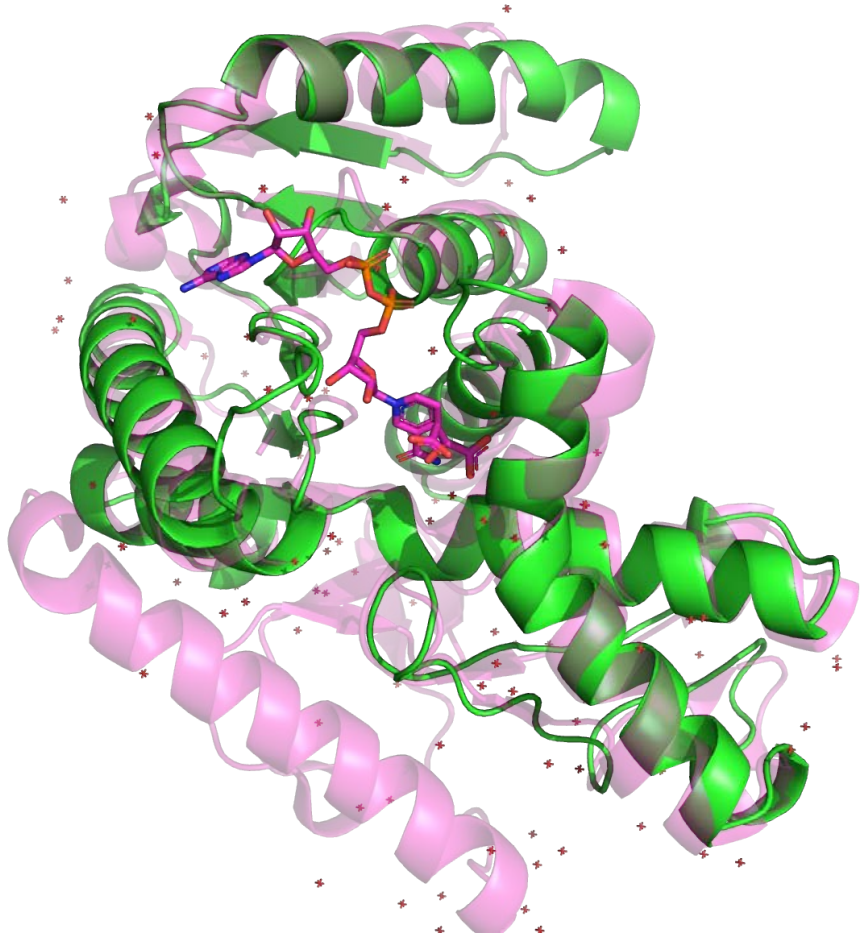
Designing enzymes by incorporating experimental feedback (aka ChatGPT for protein design)



- Need general framework that enables generative design of proteins by incorporating experimental feedback
- Genome-scale language models (GenSLMs)[1] provide a means to incorporate generative modeling for gene sequences:
  - complementary to protein language models
- Rewards for the model:
  - intrinsic – sequence specific (e.g., GC content for environmental adaptation)
  - extrinsic – functional annotation/ enzyme activity measured via experimentation

[1] M. Zvyagin, et al, Genome-scale language models map the evolutionary trajectories of SARS-CoV-2 (SC'22 Gordon Bell Prize)

Multi-objective RL for generative design allows greater sequence diversity across MDH sequences



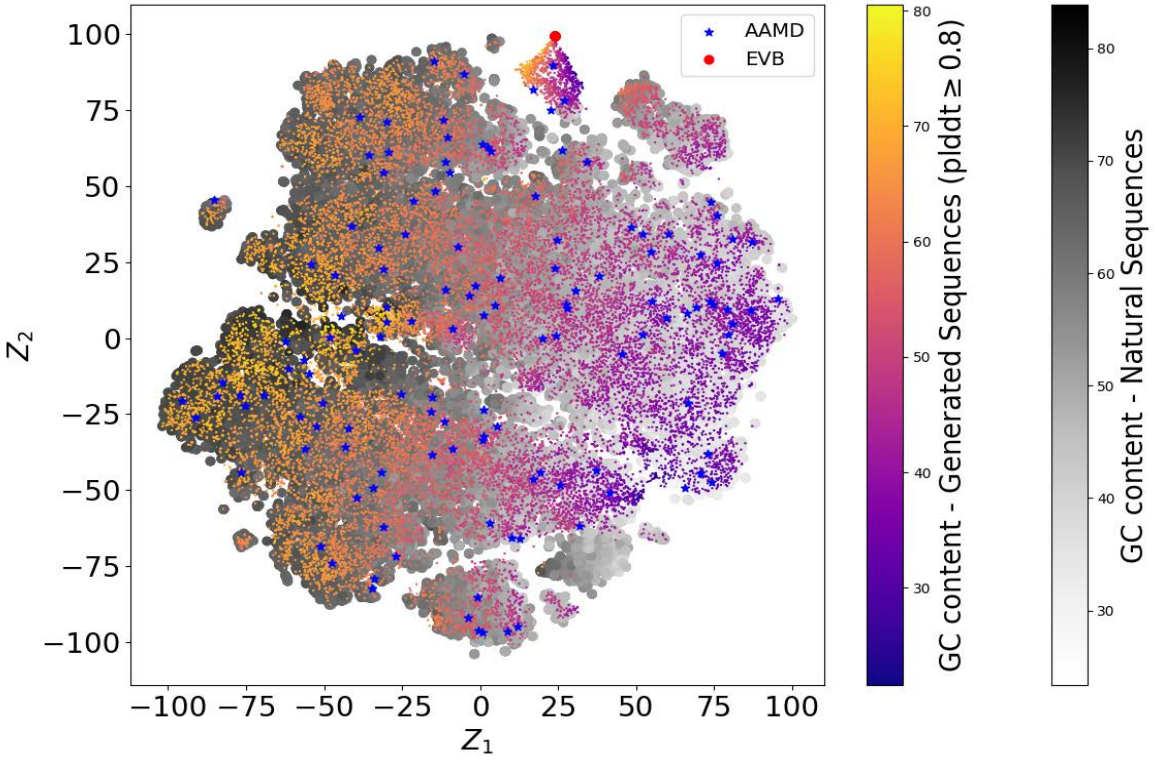
Range 17: 9 to 244 GenPept Graphics ▼ Next Match ▲ Previous Match ▲ First Match

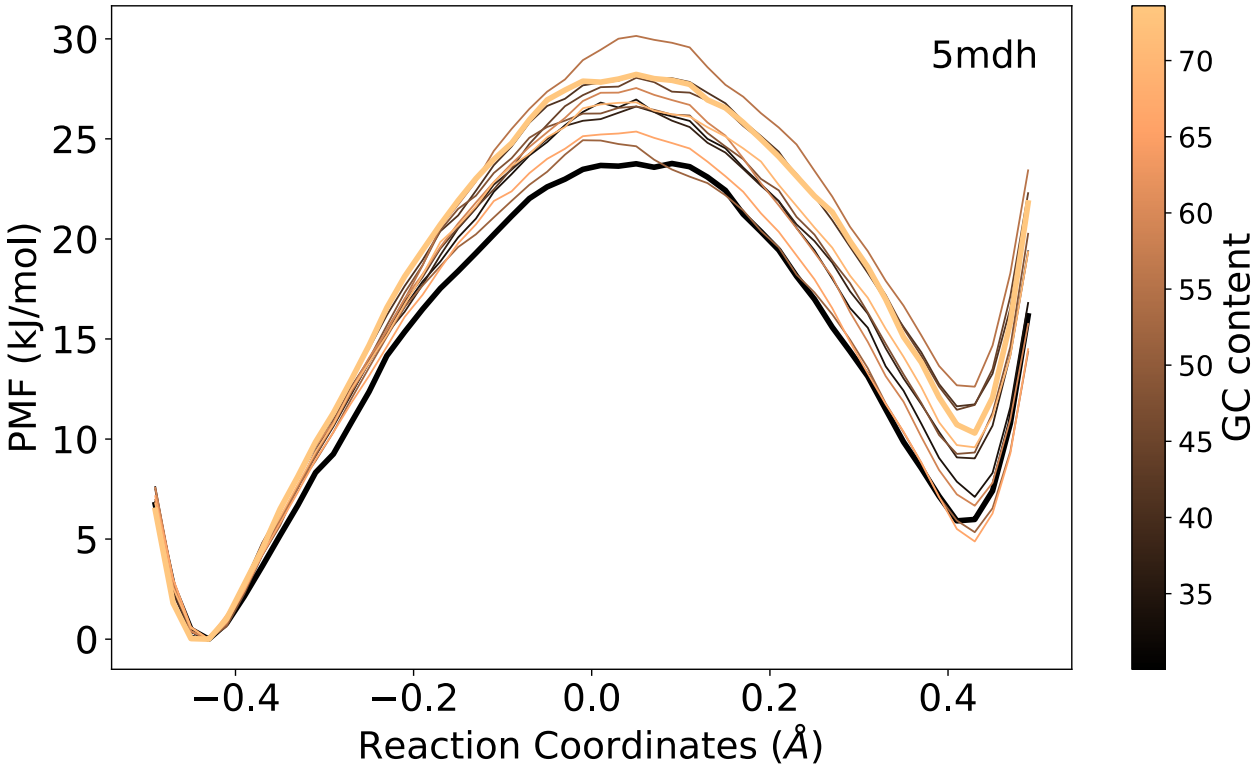
Score	Expect	Method	Identities	Positives	Gaps
89.4 bits(220)	5e-14	Compositional matrix adjust.	72/247(29%)	115/247(46%)	12/247(4%)

```
Query  9218  VAVTGAAGQIGYSLLFRIASGSMFGPDQPVVLHLIEIEPALPALQGVVMELEDCAFPLLK  9277
             + + GA G IG ++   + S  M G    + L+ I      P + GV  E++ CAFP +
Sbjct  9     LVIVGAGGMIGSNM---VQSALMLGLTPNICLYDI----FEPGVHGVFDEIQQCAFPGVN  61

Query  9278  GIVPTASLEEGFKGVNWALLVGSVPRKAGMERKELLGINGKIFVGQGKAIAANAAKDVRI  9337
              +  T + EE F G  + +  G  PRK GM R++LL  N KI    G  I     +   +
Sbjct  62    -VTYTVNPEEAFTGAKYIISSGGAPRKEGMTREDLLKGNCKIAAEFGDNIKKYCPEVEHV  120

Query  9338  LVVGNPCNTNCLIAMNNAADVPRDRWFAMTRLDENRAKAQLAKKAGVDVTTVTNMTIWGN  9397
             +V+ NP +   L A+ ++   P ++  ++  LD  R +  LA + GV    VT    +G
Sbjct  121   VVIFNPADVTALTALIHSGLKP-NQLTSLAALDSTRLQQALALEFGVQQDKVTGAHTYGG  179

Query  9398  HSATQYPDFYNAHINGRPANEV-IHDEAWLKGDFITTVQQRGAAIIKARGLSSAASAANA  9456
             H            ++G+P  E+ + DE W +    T   Q G+ IIK RG SS  S A
Sbjct  180   HGEQMAVFASQVKVDGKPLAEMGLSDERWEEIKHHTV--QGGSNIIKLRGRSSFQSPAYN  237

Query  9457  IIDTVKS  9463
              +  +++
Sbjct  238   AVKMIEA  244
```

- We can generate new sequences with varying degrees of sequence identity + positive matches
- We can also generate minimal sequences that have functional domains and can function as a productive enzyme

Why existing supercomputers may not be prepared for self-driving laboratory workflows?

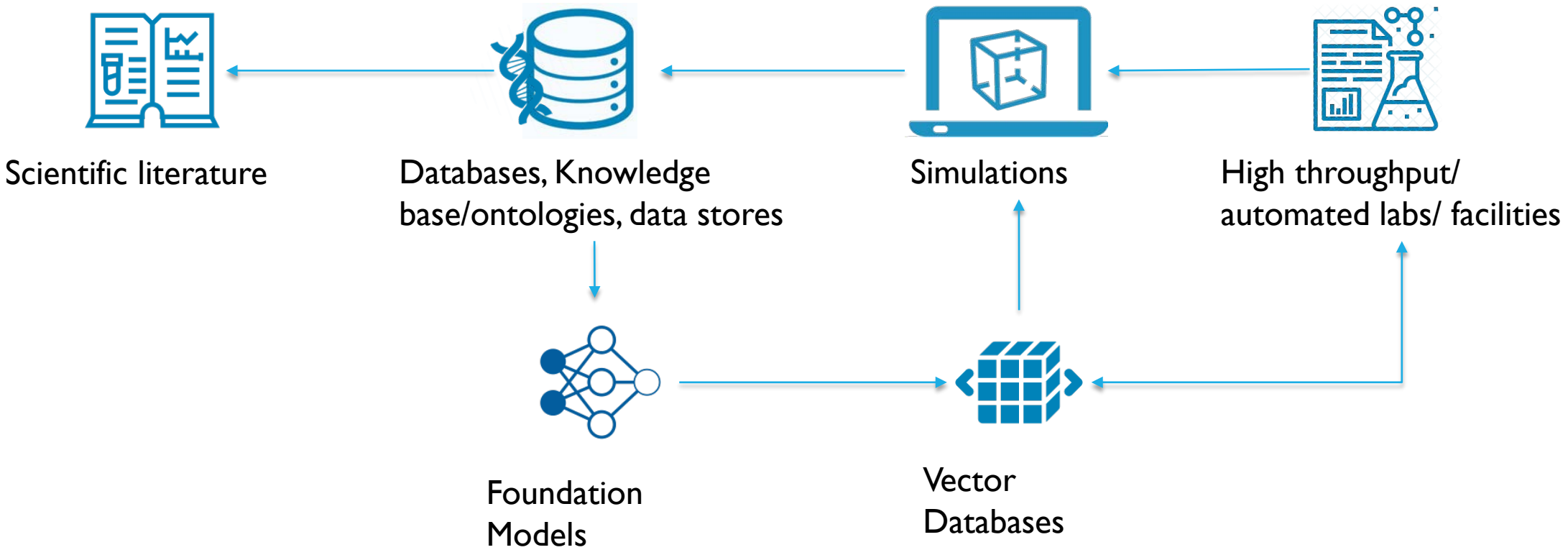




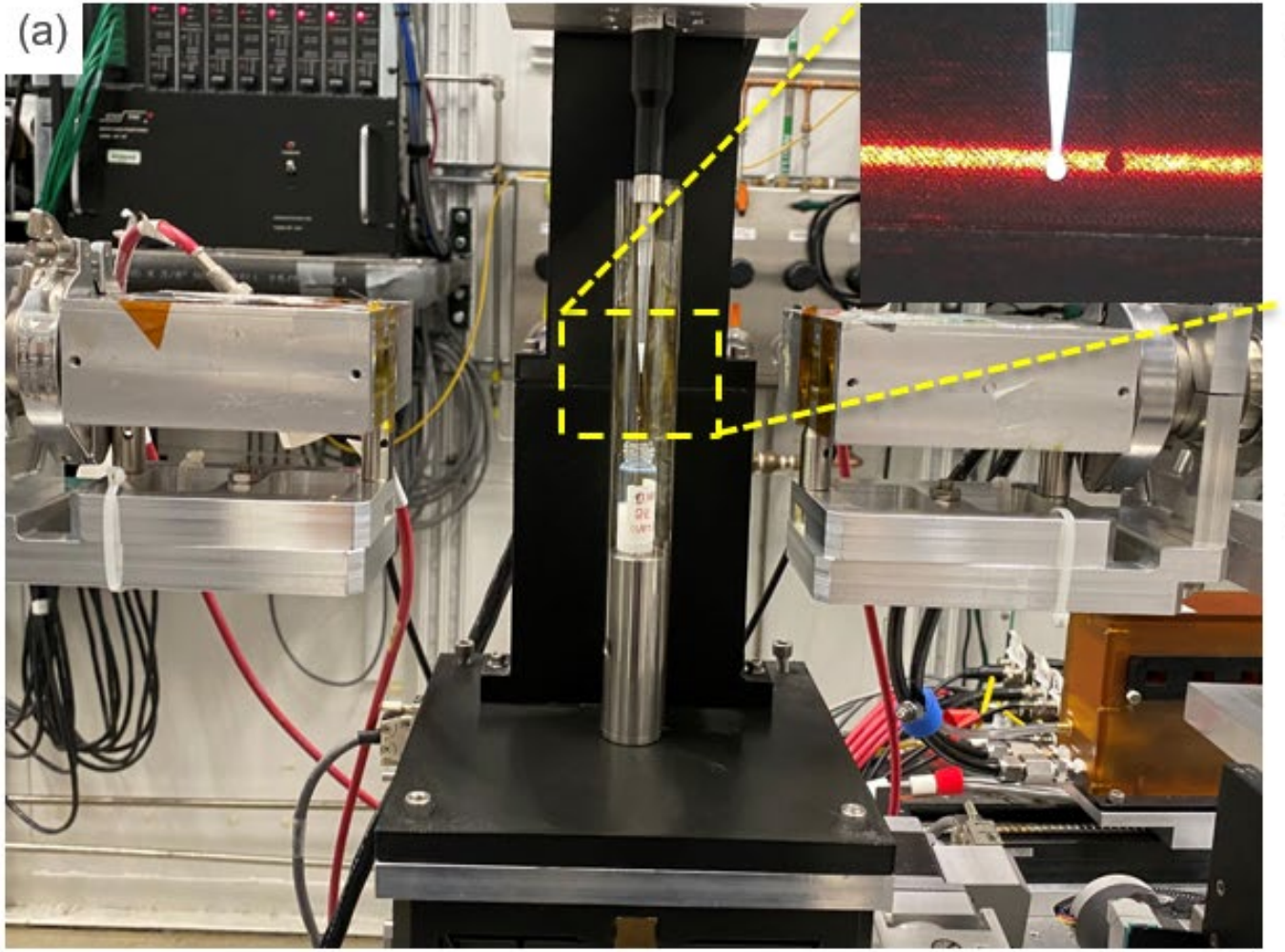
- Exploring even top 1% (1,000 variants x 20 simulation windows = 2,000 simulations) from the embedding space using simulations can overcome the limits on nodes (for a single iteration of RL-based finetuning)
- Labeling productive designs and ranking → large compute requirements across multiple computing sites/ facilities

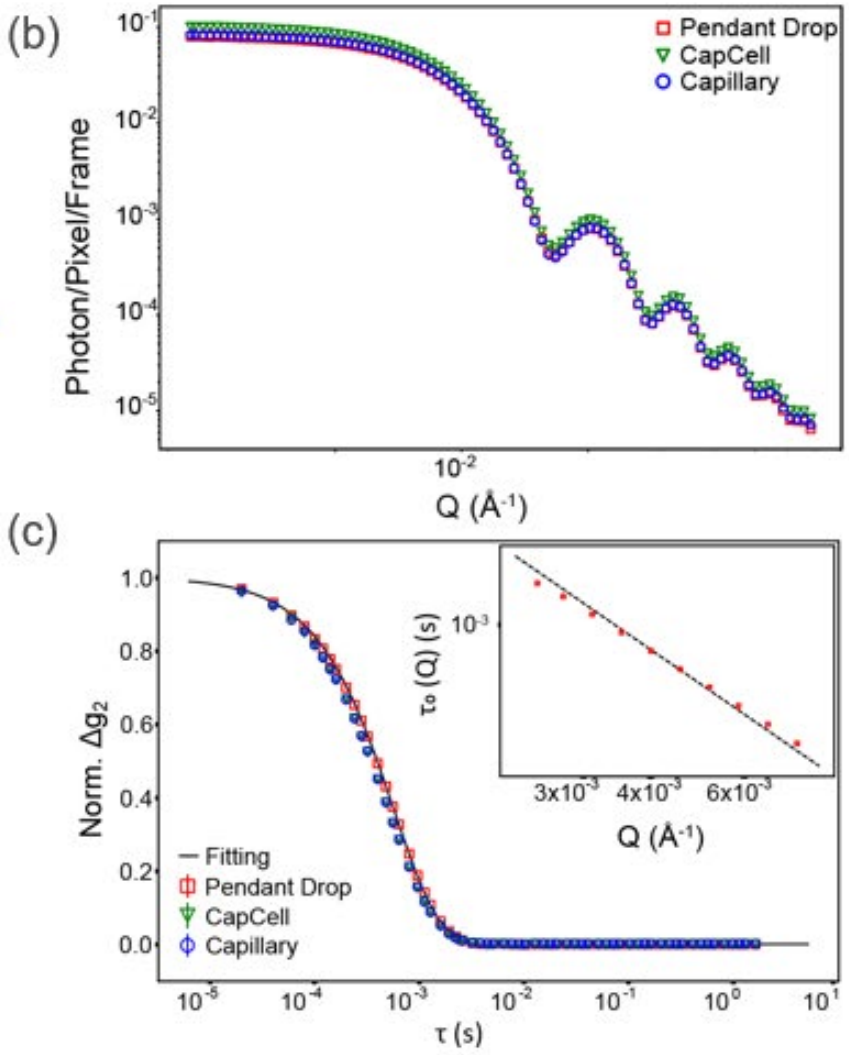
Outline

How biological data management is rapidly evolving?



We can simulate even beamlines @ APS





Summary and some take aways…

- HPC workflows are evolving to include experimental data in new and novel ways:
  - direct feedback to existing data (augmentation)
  - collection of new data to drive additional experiments
- Infrastructure support for such workflows need new thinking:
  - AI-based representation for access and relationship discovery across heterogenous datasets
  - rethinking of data fabric: experimental data, simulations, and theory within the same infrastructure
- Foundation models:
  - for empirical data, foundation models may be ideal (e.g., biology)
  - for grounded data (physics, chemistry), multimodal foundation models with theoretical underpinnings are necessary
- Automation and robotics:
  - key progress drivers; need "operating systems" that support effective integration across standards, vendors, and operators


J. Greg Paul
UChicago/ANL
Joe Insley
ANL
Vitor Mateevisti
ANL
Janet Knowles
ANL
Silvio Rizzi
ANL
Kyle H
Natalia Vassilieva
ANL
Cindy Bohorquez
Defne Ozgulbas
UIUC
Rick Stevens
Uchicago/ANL
Thomas Br
Uchicago/ANL
Rafael Vescovi
Ramanathan
Lee Liming
Vishwanath
Lukas Lacinski
ANL
Danilo Perez-Rivera
Dana
Lin
ANL
Michael E. Papka
UIC/ANL
Bharat
UIC/NIU
Doga Ozgulbus
ANL
Valerie Hayot
ANL
Abraham Stroka
ANL
Manish Shukla
ANL
Hen
ANL
Casey Stone
ANL
James
J. Davis
ANL
Michael Irvin
ANL
Zhen
Xie
ANL
Weili Nie
(NVIDIA/ Harvard)

Acknowledgements

Funding

- DOE- National Virtual Biotechnology Laboratory (NVBL)
- Exascale Computing Project Cancer Deep Learning Environment (CANDLE)
- Exascale Workflows Project (ExaWorks)
- DOE Codesign for multimodal AI
- NSF MRI: Multi-modal imaging
- DOE-MEDAL (RENEW) project for workforce training

Computing Time

- Argonne Leadership Computing (Theta/ Theta-GPU/ AI-testbed)
- Cerebras/Nvidia
- NERSC

Colleagues

- Richard Scheuermann
- James Olds
- Wesley Scott
- Anda Trifan
- Ashka Shah
- Ozan Gokdemir
- Mike Tynes

Data/ Code/ Models

- https://github.com/ramanathanlab/genslm
- Access to model weights will also be available via API

Questions/Comments

ramanathana@anl.gov