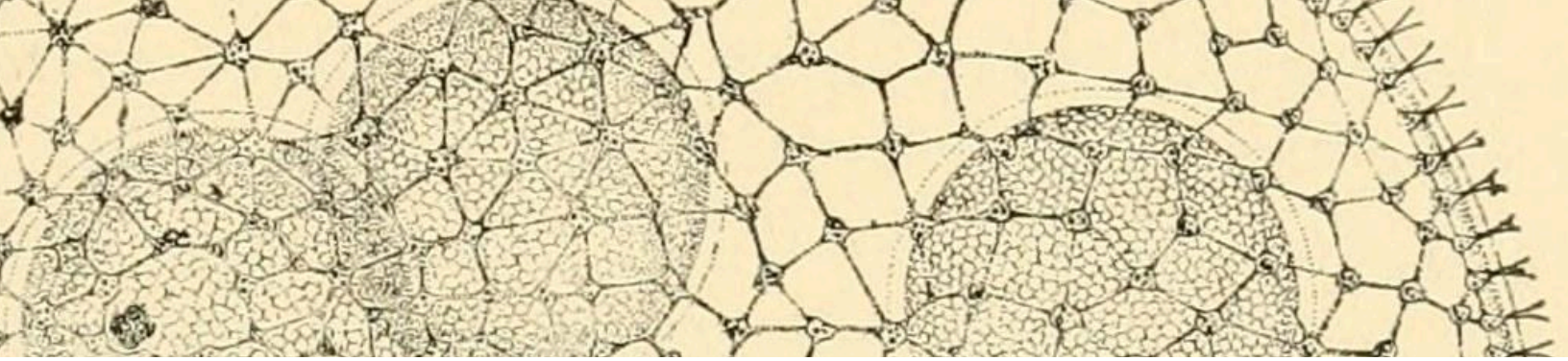

CellForge: Automating virtual cell models with multi-agent AI

Lowering barriers to building and testing models from single-cell perturbation data

Every second, billions of molecules jostle, bind, and separate inside each cell — a choreography that's nearly impossible to track in the lab. To untangle this complexity, researchers are turning to "virtual cells": computational models that simulate how real cells work. These models offer the potential for in silico experimentation, helping researchers predict cellular responses to toxins, therapeutics, and other stimuli without the need for slow, expensive laboratory work.

But while promising, developing and using virtual cell models is not a trivial task. Existing foundation models, trained on broad datasets, can underperform in contexts very different from the data they were trained on (known as "out-of-distribution" settings). Building customized models usually requires extensive interdisciplinary expertise, creating a high barrier to entry.

CellForge at a glance

- **What it is:** Multi-agent framework automating virtual cell modeling.
- **How it works:** AI agents analyze data, debate architectures, and generate runnable code.
- **Tested on:** Six perturbation datasets (gene knockouts, drugs, cytokines) across RNA-seq, CITE-seq, ATAC-seq.
- **Results:** Reduces prediction errors by up to 40%, improves correlation with experiments by ~20%.
- **Why it matters:** Makes virtual cell modeling more accessible to researchers without deep machine learning expertise.

In a new preprint, Xiangru Tang and colleagues introduce **CellForge**, a multi-agent system that automates the development of virtual cell models. When given single-cell perturbation data and task descriptions as inputs, CellForge analyzes the data, develops an appropriate model, and produces production-ready code that researchers can use for training and inference.

Approach

CellForge simulates a team of specialized experts, essentially creating a virtual lab inside a computer. Agents act like data scientists, modelers, and single-cell specialists working side by side. The framework has three modules:

- **Task analysis**: Agents assess the dataset, mine the literature, and create a research plan.

- **Design**: Domain-expert agents debate candidate architectures in a branching, graph-structured exchange before selecting a model and experimental plan.
- **Experiment execution**: Agents translate the plan into runnable code.

Benchmarking

To put the system through its paces, the team tested CellForge on six perturbation datasets spanning several common single-cell sequencing technologies. These datasets included gene knockouts, drug treatments, and cytokine stimulations, the kinds of experiments biologists routinely use to probe how cells respond to perturbations.

In every dataset tested, CellForge outperformed existing methods at predicting how cells change their gene expression after perturbation, sometimes modestly, sometimes dramatically. These gains suggest researchers can more confidently use its predictions to guide future experiments.

Performance of CellForge compared to baselines across single-cell perturbation datasets

Dataset type	Baselines tested	CellForge performance vs. baseline
Gene knockout *(scRNA-seq)*	Biolord, CondOT, CPA, scGen, scGPT	Up to ~40% lower prediction error
Gene knockout *(scCITE-seq)*	Random forest, linear regression	~2× higher correlation with experimental results
Gene knockout *(scATAC-seq)*	Random forest, linear regression	>10x higher variance explained
Drug treatment *(scRNA-seq)*	ChemCPA, CellFlow	~20% higher correlation with experimental results
Cytokine stimulation *(scRNA-seq)*	Random forest, linear regression	Clearer correlation with experimental results than simple models

Limitations

Like any system built on large language models, CellForge is not infallible. Roughly 40% of its observed failures were due to code execution issues: the sort of tensor-shape mismatches that frustrate many machine learning researchers. Its reliability also depends on how clearly the task is described, a reminder that AI pipelines still need careful human guidance.

Flexibility

One strength of CellForge lies in its flexibility: by not locking researchers into a predefined pipeline, its team of agents can design architectures tailored to the task at hand.

For example, the framework often settled on architectures similar to those an experienced scientist might choose for a particular task. Transformers tended to be favored for cytokine datasets, which require capturing long-range dependencies. In contrast, network-based methods were more effective for datasets rich in regulatory network information.

Implications

So far, CellForge has been applied only to single-cell perturbation tasks. But its flexible design could extend to other challenges, like modeling developmental trajectories, predicting cell-cell interactions, or testing how synthetic gene circuits might behave.

By lowering the barrier to virtual cell modeling, CellForge could open the door for many more labs to explore ambitious biological questions in silico — no specialized AI team required.

###