



Outerbounds

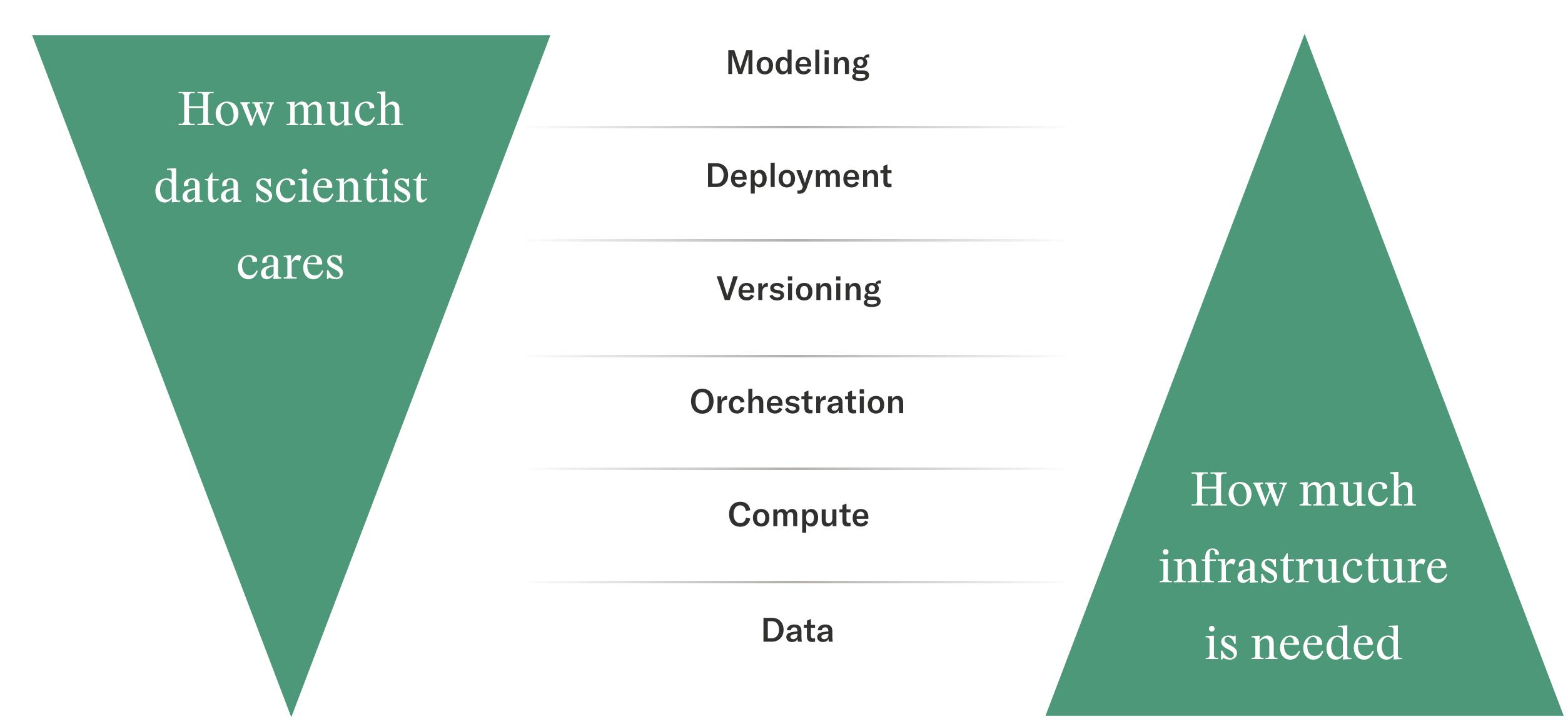
Build reproducible and scalable AI-powered computational biology systems

Consistent environments

Challenge: Computational reproducibility of notebooks.

"Out of 27,271 Jupyter notebooks, 1,203 ran through without errors, 879 (3%) that produced identical results."
<http://arxiv.org/pdf/2308.07333.pdf>

Solution: Automatic versioning of data and code dependencies.



Reproducible science

Challenge: Sharing work in teams, organizations, and with the open-source community.

Solution: Open-source development framework to version dependencies with first-class collaboration features built-in.

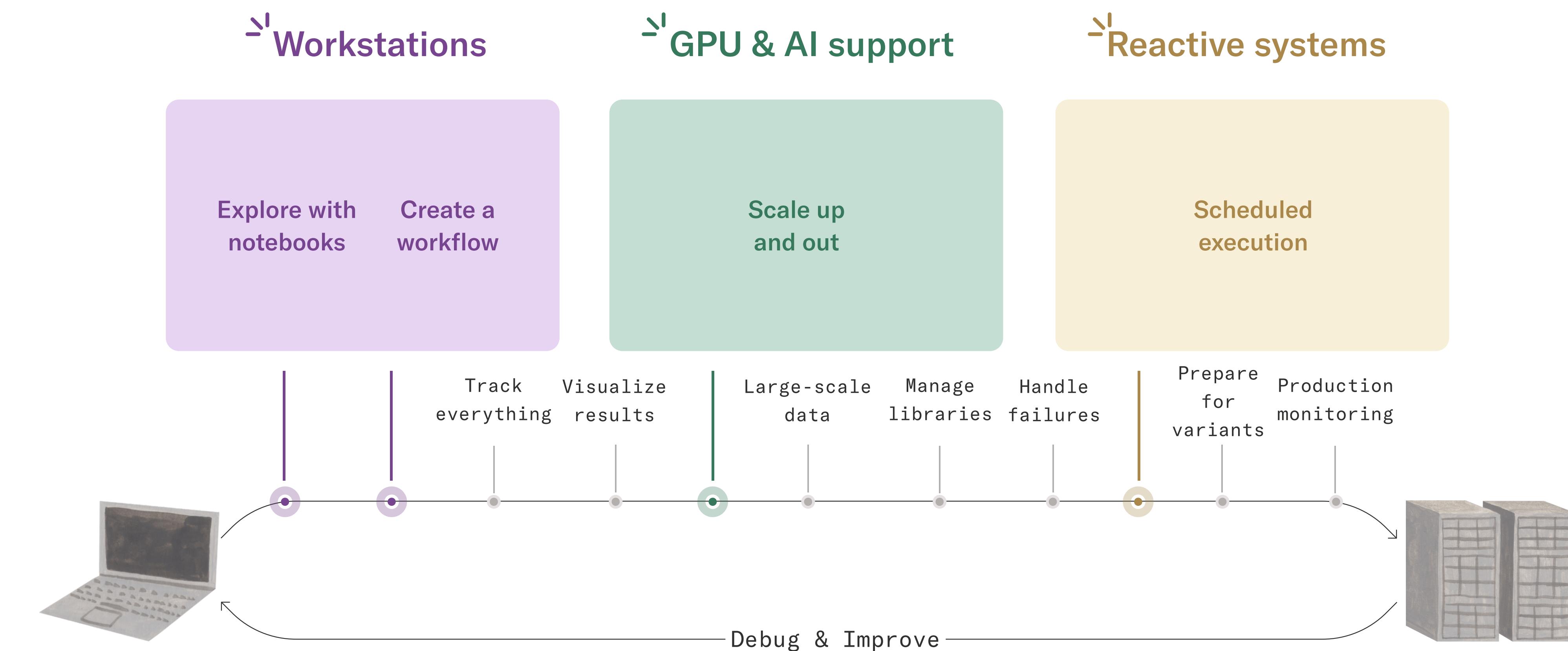


Geneformer case study

- Scale pre-training on multiple GPU nodes with DeepSpeed and Metaflow.
- Parallelize fine-tuning workflows for disease classification, cell classification, and in silico perturbation.
- Reproduce results and apply workflows to any single cell transcriptome data.

1

Explore models
in notebooks



2

Train and fine-tune on
your own data

4

Reproducible
scientific advances

3

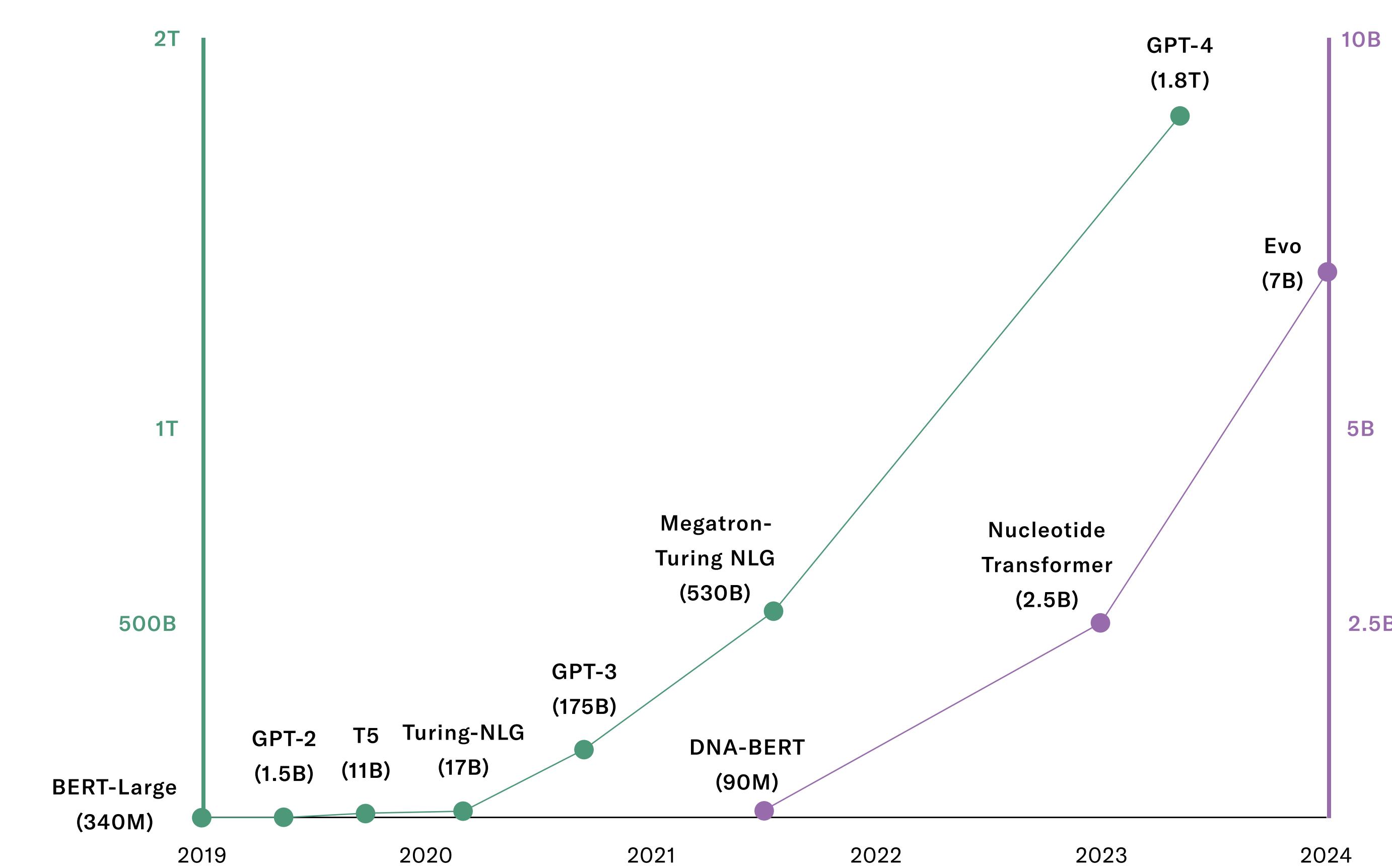
Click a button to
run the pipeline
end-to-end

Scalable compute

Challenge: Access to sufficient computational power.

Solution: Easy access to GPUs, horizontal, and vertical scalability.

Growth of transformer model size in **NLP** and **biology**



Automated workflows

Challenge: Running computational workflows automatically, on a schedule, and off your laptop.

Solution: Industry-grade workflow automation.

Reproduce state-of-the-art

- Version code, data and models
- Open-source python code
- Run on any cloud or on-prem

Compute anything

- GPUs and other accelerators
- Embarrassingly parallel tasks
- Distributed training

Automate batch jobs

- Schedule jobs
- Trigger jobs on events
- Deploy production-grade systems