

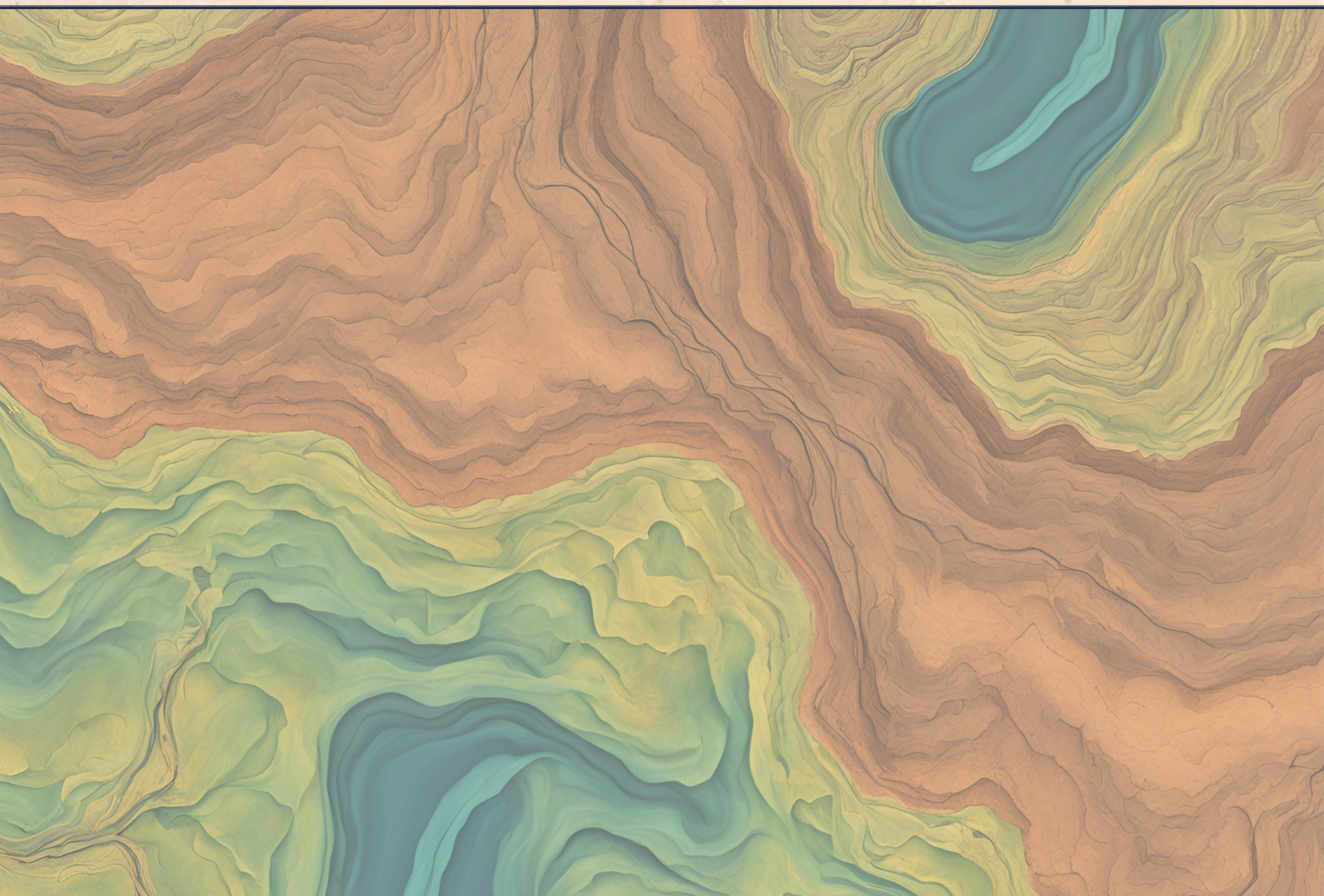


# Generative Biology On The Way

How soon until powerful AI models in biology?

An Arb Research Report

Misha Yagudin | May 2023





## Executive summary

- Forecasting technologies is difficult; for instance, recent AI progress surprised many of us. This report exploits the similarities between already-impressive Large Language Models and emerging Protein Language Models to assess the prospects of the latter. We forecast the arrival of an impressive generation in the field of biology.
- *Epistemic status*: A generalist's view; we have direct experience with text models but not protein models. This report represents 90 person-hours of work; it is not very likely to significantly change in conclusions given another 90 hours of work but conclusions might change in response to new research, in particular into scaling laws of protein language models and research into how data quality affects performance.
- *Dual-use status*: This document has gone through external review and this version is slightly edited, removing ideas beyond the current frontier. [Contact us](#) for details.
- Impressive protein generating capabilities are likely to be developed in the next 3 to 8 years. (70%)
  - The main bottlenecks are (a) data availability and quality; (b) progress on multimodal learning; and (c) learning from small datasets.
- Impressive protein generation might proliferate if (a) current open-access culture does not change; (b) licensing of the underlying proprietary datasets fails to prevent sharing model checkpoints; (c) if [structured access](#) is not implemented, allowing checks on model weights or API access. (85%)
  - So far, all of the best models have been freely available on day 1, with the papers' publication. We are not aware of any legal qualms resulting from this.

# Introduction

Progress in artificial intelligence became explosive with the deep learning (DL) revolution ([Ciresan et al 2012](#), [Krizhevsky et al., 2012](#)). Turing Award medalists [LeCun, Bengio, & Hinton \(2015\)](#) write: “These methods have dramatically improved the state-of-the-art in speech recognition, visual object recognition, object detection and many other domains such as drug discovery and genomics.”

This has culminated in products like Midjourney ([Midjourney, 2022](#)) and ChatGPT ([OpenAI, 2022](#)), models that can produce human-like output from simple instructions to consumers. These models are called *generative* AI because they can generate novel data that captures relevant patterns and to some extent generalizes from the input data. Further, such models can often be flexibly conditioned to produce data with very specific desired properties (style transfer, format specifications, etc).

These models proliferated fast. Competitive open-source models for text-to-image generation become widely accessible within 18 months of the original proprietary models, and within 4 months for conversational AI (see section [Rapid proliferation through open-source](#)).

This dynamic is worrying for more threatening domains. Already we see that AI techniques presently used for drug-discovery can be easily repurposed to find dangerous substances ([Urbina et al., 2022](#)). Further, large language models also produce undesirable “toxic” content despite intense efforts to make them harmless ([Glaese et al, 2022](#), [Bender et al., 2021](#)), and we have no reliable countermeasure to “prompt injection” attacks which completely sidestep task conditioning and prosocial finetuning ([Greshake et al 2023](#)).

The question arises: when will generative AI for drug discovery be similarly impressive? (e.g., reducing the test burden such that a nonspecialist can generate molecules with their chosen properties.) Coupled with the history of broad proliferation and availability of cloud labs ([Dunlap & Pauwels, 2017](#); [National Academies of Sciences, Engineering and Medicine, 2018](#))<sup>1</sup>, this could empower malicious actors to cause great harm ([Brockmann et al., 2019](#); [Ekins et al., 2023](#)).

Already Ferruz et al. ([2023](#)) show that “generative sequence models can be employed to create large libraries of plausible sequences for which oracle models predict structure and function properties in a matter of hours. This process requires no technical DL expertise and provides data from which experts can select and refine to one or a set of promising candidates that constitute starting points for in-vitro experiments.”

“Drug discovery” is a very broad field, so we limit ourselves to *in vivo* protein design and in

---

<sup>1</sup> Combining AI with experimentation in physical labs would likely be bottlenecked by labs’ throughput ([Crécy-Lagard, et al., 2022](#)) but digital copies of physical labs (or digital twins ([Tao & Qi, 2019](#)) could mitigate that and have already been proven to be useful in boosting manufacturing ([Ferruz et al., 2023](#)).

particular *language modeling techniques for protein sequence modeling*, as this is the most direct comparison with generative AI progress in the text domain, thanks to the similarities between natural language and protein language.

To answer this question we will discuss drivers and characteristics of AI progress; we decompose the trajectory of AI solving a domain into two steps:

1. an “ImageNet moment” (“a model, dataset and pretraining task that provide strong off-the-shelf performance for most tasks, even with little data” to quote [Ofer et al. 2021](#)), followed by
2. impressive<sup>2</sup> generative models first being trained.

We use this two-step rubric to guess when biology’s “ImageNet moment” will arrive and how long it will take to develop “impressive” generative models following it. Finally, we guess how fast these models will proliferate under a business-as-usual scenario.

---

<sup>2</sup> Here “impressive” is of course imprecise, we are roughly tracking “widely used by the general public because of how good it is.”



# What drives AI progress? How is it measured?

To predict AI progress, it's useful to get context on how it progressed in the past. Specifically, what was the course of:


- progress in computer vision (CV), starting with AlexNet and culminating in text-to-image synthesis models like DALL-E ([Ramesh et al., 2021](#)); and
- progress in “deep” natural language processing (NLP) starting with early applications to classical NLP tasks, s.a. machine translation ([Sutskever et al., 2014](#), [Bahdanau et al., 2015](#)) and culminating in chat-bot systems like ChatGPT ([OpenAI, 2022](#)).

## Computer vision: AlexNet to DALL-E

Despite their different objectives and architectures, here we place image classification and image generation in the same lineage. We use the dataset of ML systems collected by Sevilla et al. ([2021](#)) and curate a subset of notable systems (see page 6).

A rough summary of the following table: in late 2012, convolutional neural networks showed promise. Shortly afterwards, systems pre-trained on ImageNet, such as DeCAF, showed success in all sorts of CV tasks; transfer learning by pretraining on ImageNet was so effective that it became the default ([Mahajan et al., 2018](#)). **Image generation progressed from underwhelming to very impressive in 7 years.** In mid 2022, a text-to-image model was open-sourced by StabilityAI ([2022](#)), which stimulated a lot of further open-source progress ([Willison, 2022](#)).

A comprehensive study of benchmark dynamics ([Ott et al., 2022](#)) roughly agrees with our characterisation of progress: “In computer vision, high research intensity and continuous progress on image classification benchmarking (Supplementary Fig. 1) started in 2013. This is earlier than most other AI tasks, as those were the first application areas in which deep learning started to excel. Notable later advances happened in 3D vision processing (since 2016), image generation (since 2017) and few-shot learning (2018–2019). In terms of relative SOTA improvements, the map for CV shows a wide array of patterns in benchmark dynamics across different AI tasks that elude simple narratives about benchmark intensity and progress

AlexNet	30/09/2012	<a href="#">Krizhevsky et al., 2012</a>	Early DL system for image classification starts "deep learning revolution"
DeCAF	04/11/2013	<a href="#">Donahue, et al., 2013</a>	First system pre-trained on ImageNet achieve SoTA results on some CV tasks.
VAE	20/12/2013	<a href="#">Kingma &amp; Welling, 2013</a>	Early DL image generation system
GANs	10/06/2014	<a href="#">Goodfellow et al., 2014</a>	Early DL image generation system
			GANs progress in face generation ( <a href="#">Besiroglu, 2021</a> ): <a href="#">GANs</a> , <a href="#">DCGANs</a> , <a href="#">CoGANs</a> , <a href="#">ProGAN</a> , <a href="#">StyleGAN</a> (and modifications: <a href="#">on two</a> )
DALL-E	05/01/2021	<a href="#">Ramesh et al., 2021</a>	Early advanced text-to-image system
DALL-E 2	06/04/2022	<a href="#">Ramesh et al., 2022</a>	Advanced text-to-image system, API made available by OpenAI Nov 2022
Stable Diffusion	13/04/2022	<a href="#">Rombach et al., 2022</a>	Advanced text-to-image system openly released to the public in Aug 2022

## NLP: Seq2Seq to ChatGPT

We use the dataset of machine learning systems collected by Sevilla et al. ([2021](#)) and curate a subset of notable ML systems:

NMT	01/09/2014	<a href="#">Bahdanau et al., 2015</a>	Early DL machine translation system
Seq2Seq	10/09/2014	<a href="#">Sutskever et al., 2014</a>	Early DL machine translation system
Transformer	12/06/2017	<a href="#">Vaswani et al., 2017</a>	Architectural advance
ULM-FiT	18/01/2018	<a href="#">Howard &amp; Ruder, 2018</a>	Early “NLP ImageNet moment”
ELMo	01/02/2018	<a href="#">Peters et al., 2018</a>	Early “NLP ImageNet moment”
GPT	01/06/2018	<a href="#">Radford et al., 2018</a>	Early “NLP ImageNet moment”
GPT-2	14/02/2019	<a href="#">Radford et al., 2019</a>	Extreme transfer learning from language modelling task
GPT-3	28/05/2020	<a href="#">Brown et al., 2020</a>	Demonstrating power of scaling
InstructGPT	04/03/2022	<a href="#">Ouyang et al., 2022</a>	Making GPT-3 much more useful through human-feedback
ChatGPT	30/11/2022	<a href="#">OpenAI, 2022</a>	Chatbot based on InstructGPT ideas achieve wide adoption
LLaMA	27/02/2023	<a href="#">Touvron et al., 2023</a>	Leaked LLM, leading to mass proliferation

Rough summary: in late 2014, deep neural networks were first applied to classical tasks in NLP. In 2017, the Transformer architecture was developed, enabling large pre-trained language models. In early-mid 2018, “NLP’s ImageNet moment arrived” ([Ruder, 2018](#)), pretrained language models were used to achieve state-of-the-art results on a wide range of NLP tasks, following the success of CV models pre-trained on ImageNet. OpenAI scaled their Generative Pre-trained Transformer (GPT), showing remarkable capability advancement. They later fine-tuned GPT-3 to be particularly useful, leading to the wide adoption of ChatGPT in Dec 2022. In March 2023, LLaMa (a large language model developed by Meta) leaked, enabling mass proliferation ([Willison, 2023](#)).



**In 4–5 years, NLP went from its ImageNet moment to the widespread and diffusion of advanced generative models which can be run by any technically literate user.**

A comprehensive study of benchmarks agrees with our story ([Ott et al., 2022](#)): “In NLP the tasks of information extraction, sentiment analysis, language modeling and question answering had significant density of novel SOTA results the earliest (2014–2016). It is noteworthy that none of the tasks completely ceased to produce SOTA activity once they became established. Relative SOTA improvements were modest until 2018. There was a slight clustering of large relative SOTA improvements around 2018–2019—a possible interpretation being that this was when AI language capabilities experienced a boost while benchmarks were not yet saturated.”

## What explains AI progress?

Richard Sutton ([2019](#)) summarizes the key insight from the 70-year history of AI research as “general methods that leverage computation are ultimately the most effective, and by a large margin.” Within the deep learning paradigm, we notice a further pattern:

- *Key architecture found*: an expressive general architecture is discovered and improved (CNNs, Transformers);
- *ImageNet moment*: it is trained at scale on a large dataset, producing a general-purpose model (for that data domain) which generalizes to other tasks, or can be used as a backbone for task-specific models through transfer learning;
- *Scaling up*: models are scaled following compute-optimal recipes derived from scaling laws, more and more impressive capabilities are achieved including more and more powerful generalization;
- *Post-training*: advanced evaluation techniques and fine-tuning are used to make a very capable model particularly useful.

This has been retroactively named the “Foundation Model” paradigm ([Bommasani et al., 2021](#)).

## AI Triad

Buchanan ([2020](#)) reduces the complexities of modern AI to three elements: “machine learning systems use *computing* power to execute *algorithms* that learn from *data*.” This “AI Triad” is the availability of computing power to scale models<sup>3</sup>, advances in algorithms to effectively learn representations<sup>4</sup>, and the availability of data used to implicitly specify patterns to learn<sup>5</sup>. These are the key drivers of performance of modern ML.

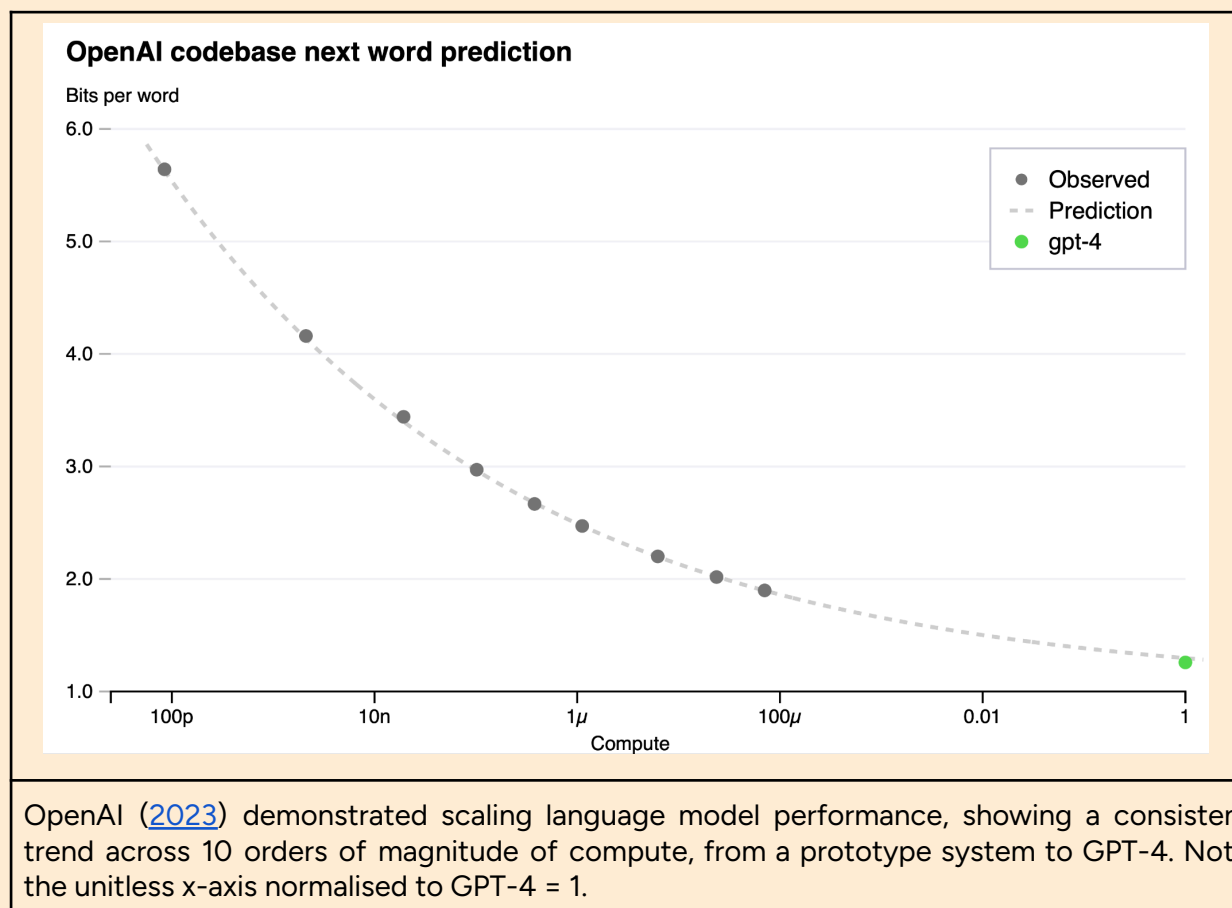
<sup>3</sup> Compute scaling references: Hestness et al. ([2017](#)), Amodei & Hernandez ([2018](#)), Bartoldson et al. ([2023](#))

<sup>4</sup> Algorithmic improvement references: Hernandez & Brown ([2020](#)) and Erdil & Besiroglu ([2022](#))

<sup>5</sup> Data scaling references: Ré ([2021](#)) and Villalobos & Ho ([2022](#))

AI Triad abstracts key input to machine learning progress<sup>6</sup>. AI Triad is interdependent — scaling laws describe how much data, compute, and parameters one needs to achieve a desired performance using given architecture.

## Scaling Laws



As models have grown in size and complexity, researchers have retroactively fit empirical scaling “laws” that govern their performance. These curves show that the loss of a transformer language model tends to improve as the amount of compute used to train the model increases following a predictable pattern<sup>7</sup>.

According to Villalobos (2023), “the modern study of scaling laws arguably started with Hestness et al. (2017), who empirically identified power-law scaling of the test loss with

<sup>6</sup> E.g., advancement in hardware feeds into compute availability; research and engineering talent feeds into algorithmic progress; development of benchmarks feeds into algorithmic progress as benchmarks guide research direction as “what you measure is what you get.”

<sup>7</sup> One needs to note, that while “loss” follows a predictable pattern, to our knowledge, there is no predictable understanding of when human-understandable capabilities emerge in terms of “error.” See Gwern (2020) and Wei et al. (2022) for a discussion of emergent capabilities. Bowman (2023) writes “GPT-3’s capacity for few-shot learning on practical tasks appears to have been discovered only after it was trained, and its capacity for chain-of-thought reasoning was discovered only several months after it was broadly deployed to the public.”

respect to training data size in several different domains. In Hestness et al. ([2019](#)) this previous result was used to predict the increases in model and dataset sizes that would be needed to reach important performance milestones.” And later on Kaplan et al. ([2020](#)), Hoffmann et al. ([2022](#)), OpenAI ([2023](#)), and Google ([2023](#)) further verified them at much larger scales.

Villalobos ([2023](#)) concludes: “while ‘scale is all you need’ seems mostly true for direct training, when it comes to transfer learning, the downstream performance critically depends on the tasks at hand as well as the choice of architecture and hyperparameters ([Tay et al., 2022](#)). When the upstream and downstream tasks are similar, downstream loss can be reasonably well predicted from upstream loss, but this is not the case when the two tasks are substantially different ([Abnar et al., 2021](#)).”

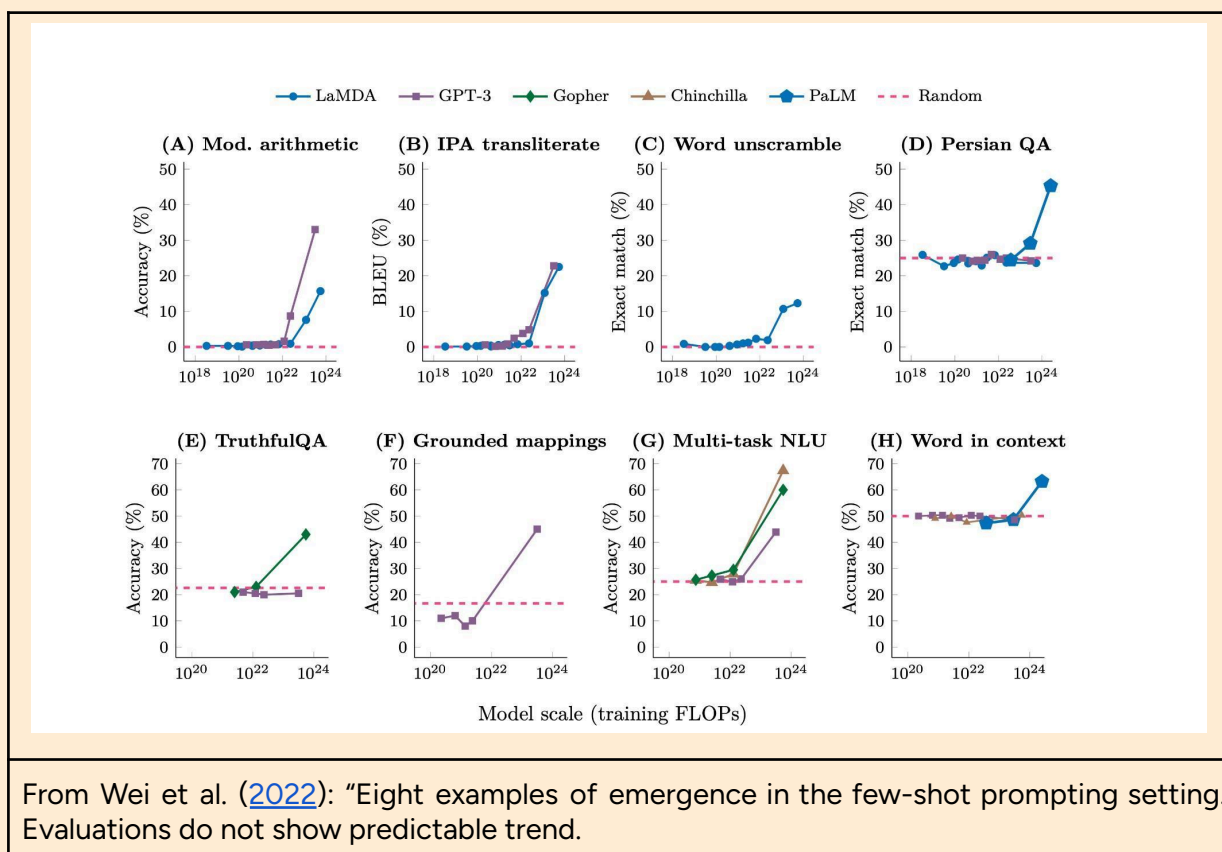


## Other characteristics of AI progress

### AI progress surprised us

Progress in AI and applications surprised almost everyone, from highly regarded AI scientists to professional forecasters and subject matter-experts; a further paper examining 3765 benchmarks concludes that “progress in AI as captured by improvements in SOTA benchmark results remains rather unpredictable and prone to unexpected bursts of progress” ([Ott et al., 2022](#)). Bowman ([2023](#)) notes that while “LLMs predictably get more capable with increasing investment, even without targeted innovation” (see our [section on scaling laws](#)), “[m]any important LLM behaviours emerge unpredictably as a byproduct of increasing investment.”

The sentiment is shared by Ganguli et al. ([2022](#)): “generative models have a paradoxical combination of predictable loss on a broad training distribution (as embodied in their “scaling laws”), and unpredictable specific capabilities, inputs, and outputs. We believe that the high-level predictability and appearance of useful capabilities drives rapid development of such models, while the unpredictable qualities make it difficult to anticipate the consequences of model deployment.”



- Advances in artificial intelligence happened faster than was commonly anticipated, Geoffrey Hinton, sometimes referred to as a “Godfather of Deep Learning” summarized it in a recent interview with CBS ([2023](#)): “Until quite recently, I thought it was going to be like 20 to 50 years before we had general purpose AI. And now I think it may be 20 years or less.” and “I wouldn't completely rule that possibility [having general purpose AI in five years] out now. Whereas a few years ago I would have said no way.”
- In 2021, Jacob Steinhardt’s research group commissioned 6 AI questions for professional forecasters ([Steinhardt, 2021](#)). In his one-year retrospective he writes: “progress on ML benchmarks happened significantly faster than forecasters expected. But forecasters predicted faster progress than I did personally, and my sense is that I expect somewhat faster progress than the median ML researcher does” ([Steinhardt, 2022](#)). Further, we can already note that GPT-4 ([OpenAI, 2023](#)), released in early 2023, already exceeds the forecast for 2025 (on the one measure which OpenAI reported).
- In biology, DeepMind’s AlphaFold shattered the competition in the CASP13 tournament: “an anomalous leap, on the order of a doubling of the usual rate of improvement” ([AlQuraishi, 2018](#)). Two years later, AlphaFold2 was an “advance so thorough it compelled CASP organizers to declare the protein structure prediction problem for single protein chains to be solved”, to quote AlQuraishi ([2020](#)). This went beyond theoretical significance, according to McKinsey & Company ([2022](#)): “Biopharma internalized AlphaFold2 and ColabFold to generate 3-D models of almost any known, synthesized protein and protein–protein interactions, reducing access to 3-D structures from 6 months to a few hours.”
- Ott et al. ([2022](#)) curated “3765 benchmarks covering the entire domains of computer vision and natural language processing” and found that “dynamics of performance gains on specific AI tasks usually do not follow clearly identifiable patterns. This indicates that progress in AI as captured by improvements in SOTA benchmark results remains rather unpredictable and prone to unexpected bursts of progress and phases of saturation/stagnation. This is likely caused both by the complexities and limitations of current benchmarking practices, as well as actual sudden bursts in AI capabilities.”

## Rapid proliferation through open-source

The ML community has extremely strong openness norms. Developers of AI systems often open source them, including publicly sharing the model's weights, code, and the datasets used to train the model. Such norms are valuable in any science, allowing researchers to reproduce and build on each other's work.

At the same time, given growing concerns about malicious use of AI ([Brundage et al., 2018](#)), other approaches such as *structured access* have emerged ([Shevlane, 2022](#)). Early attempts, like the staged release of GPT-2 by OpenAI ([Radford et al., 2019](#) and [Solaiman et al., 2019](#)) faced harsh criticism from the community for not open sourcing their model ([Zhang, 2019](#)) and the decision to not release GPT-3 code ([Microsoft, 2020](#)) faced similar criticism ([Riedl, 2020](#)).

Today, more organizations limit access to their models (as both weights and structured access). But many highly capable models are still open-sourced. Early on T5-11B ([Raffel et al., 2019](#)) open-source by Google in early 2020 was the most powerful publicly available mode. A somewhat capable model, Meta's OPT-175B was released in early 2022, for non-commercial use only ([Zhang et al., 2022](#)). This was followed by the full open-sourcing of BLOOM, a 176B parameter model developed by BigScience ([BigScience, 2022](#)). The volunteer group EleutherAI has open-sourced several smaller models: GPT-Neo-2.7B in early 2020 ([Black et al., 2020](#)), GPT-J-6B in mid-2021 ([Wang & Komatsuzaki, 2021](#)), GPT-NeoX-20B in early 2022 ([Black et al., 2022](#)), and their Pythia training is in progress ([EleutherAI, 2023](#)). Another mode of open-sourcing involves fine-tuning an open-source model on outputs from a closed-source model, a kind of slow and lossy model extraction attack ([Chavinlo, 2023](#)). Most recently, we can examine the proliferation of generative vision AI models similar to DALL-E. We see that the open source community cannot yet match frontier models like GPT-4<sup>8</sup>, but it has had no problem catching up<sup>9</sup> (and sometimes outpacing) models such DALL-E, sometimes surpassing massive industrial labs when a good pretrained model is made available to them.



Most recently, we can examine the proliferation of generative vision AI models similar to DALL-E. We see that the open source community cannot yet match frontier models like GPT-4<sup>8</sup>, but it has had no problem catching up<sup>9</sup> (and sometimes outpacing) models such as DALL-E, sometimes surpassing massive industrial labs when a good pretrained model is made available to them.

As a result we are not able to ensure safeguards are employed in releasing models. For example, the infamous “GPT-4chan” was trained by fine-tuning a previously open-sourced model GPT-J on a notably bigoted internet corpus ([Kurenkov, 2022](#)). Another example, pointedly summarized by Wikipedia ([2023](#)): “Auto-GPT was used to create ChaosGPT, which, given the goal of destroying humanity, was not immediately successful in doing so.”

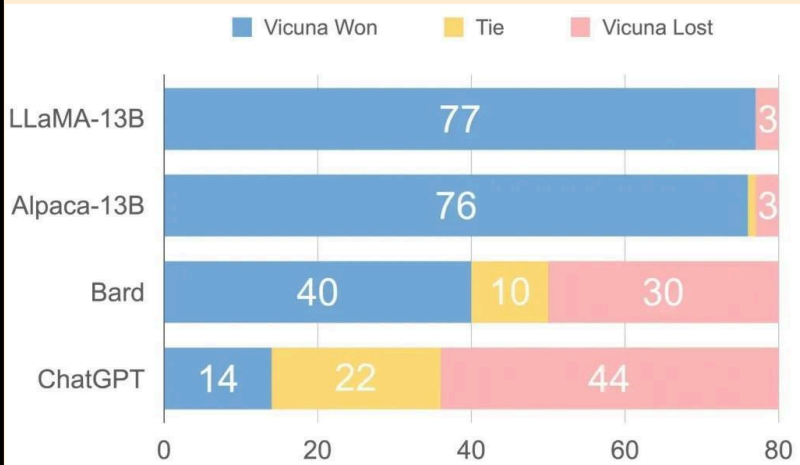
### Timeline of proliferation in text-to-image generation

January 2021	DALL-E, the first advanced text-to-image generation system announced by OpenAI ( <a href="#">OpenAI, 2021</a> )
April 2022	DALL-E 2, the follow up system announced by OpenAI ( <a href="#">OpenAI, 2022</a> )
April 2022	Stable Diffusion announced ( <a href="#">stability.ai, 2022</a> )
May 2022	Imagen announced by Google ( <a href="#">Saharia et al., 2022</a> )
May 2022	DALL-E available through waitlist
July 2022	<a href="#">Craiyon</a> (DALL-E Mini) open-sourced under Apache 2.0
July 2022	<a href="#">Midjourney</a> , an advanced proprietary model announced
August 2022	StabilityAI open-sources Stable Diffusion under Creative Commons OpenRAIL-M ( <a href="#">stability.ai, 2022</a> )
September 2022	DALL-E available without waitlist ( <a href="#">OpenAI, 2022</a> )
November 2022	OpenAI launches DALL-E API for developers ( <a href="#">OpenAI, 2022</a> )


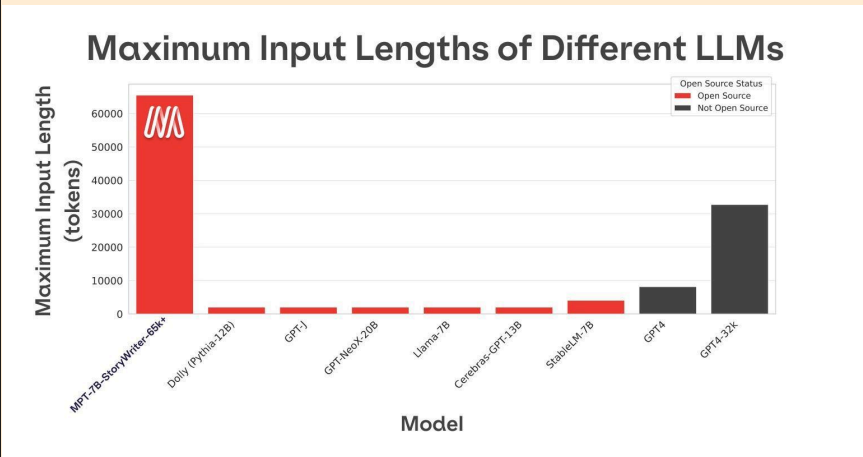
<sup>8</sup> While writing this report, BLOOMChat was open-sourced, achieving a 45% win-rate against GPT-4 in a human preference study ([SambaNova, 2023](#)), making the gap between open-source and proprietary models even smaller.

<sup>9</sup> After finishing this report, Gudibande et al. ([2023](#)) published a more thorough evaluation of ChatGPT-imitating models (like Alpaca ([Taori et al., 2023](#))), finding that “imitation models close little to none of the gap from the base LM to ChatGPT on tasks that are not heavily supported in the imitation data” and “these performance discrepancies may slip past human raters because imitation models are adept at mimicking ChatGPT’s style but not its factuality.” This indicates that proprietary models potentially have an advantage despite claims like “We [Google] Have No Moat, And Neither Does OpenAI” ([Semianalysis, 2023](#)).

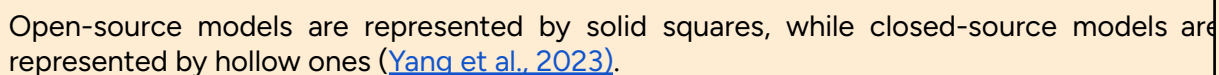
## Timeline of proliferation in text-to-text generation

January 27, 2022	InstructGPT a fine-tuning technique powering ChatGPT is announced ( <a href="#">OpenAI, 2022</a> )																				
November 31, 2022	ChatGPT is released by OpenAI, it's an overnight success amassing 1 users in 5 days and 100M users in 2 months ( <a href="#">OpenAI, 2022</a> ).																				
February 6, 2023	Bard is announced by Google ( <a href="#">Google, 2023</a> )																				
February 24, 2023	LLaMa, a large language model, is released by Meta; it is available to researchers for non-commercial purposes only ( <a href="#">Meta AI, 2022</a> )																				
March 3, 2023	LLaMa was leaked ( <a href="#">The Verge, 2023</a> )																				
March 13, 2023	Alpaca-7B, an instruction-fine tuned LLaMa-7B, is released by Stanford at a fine-tuning cost of 600 USD ( <a href="#">Taori et al., 2023</a> ).																				
March 14, 2023	GPT-4 is released by OpenAI and is integrated into ChatGPT ( <a href="#">OpenAI, 2023</a> )																				
March 14, 2023	Claude is released by Anthropic ( <a href="#">Anthropic, 2023</a> )																				
March 21, 2023	Google opened up early access for Bard via a waitlist ( <a href="#">Google, 2023</a> )																				
March 30, 2023	<p>Vicuna-13B, an instruction-fine tuned LLaMa-13B, is released by Chiang et al. at a fine-tuning cost of 300 USD (<a href="#">The Vicuna Team, 2023</a>).</p>  <table><thead><tr><th></th><th>Vicuna Won</th><th>Tie</th><th>Vicuna Lost</th></tr></thead><tbody><tr><td>LLaMA-13B</td><td>77</td><td>0</td><td>3</td></tr><tr><td>Alpaca-13B</td><td>76</td><td>0</td><td>3</td></tr><tr><td>Bard</td><td>40</td><td>10</td><td>30</td></tr><tr><td>ChatGPT</td><td>14</td><td>22</td><td>44</td></tr></tbody></table>		Vicuna Won	Tie	Vicuna Lost	LLaMA-13B	77	0	3	Alpaca-13B	76	0	3	Bard	40	10	30	ChatGPT	14	22	44
	Vicuna Won	Tie	Vicuna Lost																		
LLaMA-13B	77	0	3																		
Alpaca-13B	76	0	3																		
Bard	40	10	30																		
ChatGPT	14	22	44																		
April 1, 2023	GPT4-x-Alpaca, a LLaMA 13B model fine-tuned with a collection of GPT4 conversions, is released ( <a href="#">Teknium, 2023</a> ).																				

<sup>10</sup> For another set of evaluations, see the Open LLM Leaderboard ([HuggingFace, 2023](#)) and Chatbot Arena ([LMSYS, 2023](#)).

April 24, 2023	<p>WizardLM, a fine-tuned 7B LLaMA mode, is released (<a href="#">Xu et al., 2023</a>).</p>  <table><tr><th>Model</th><th>Ours Win (%)</th><th>Ours Lose (%)</th><th>Tie (%)</th></tr><tr><td>Alpaca-7B</td><td>41</td><td>16</td><td>6</td></tr><tr><td>Vicuna-7B</td><td>30</td><td>21</td><td>12</td></tr><tr><td>ChatGPT</td><td>27</td><td>22</td><td>14</td></tr></table>	Model	Ours Win (%)	Ours Lose (%)	Tie (%)	Alpaca-7B	41	16	6	Vicuna-7B	30	21	12	ChatGPT	27	22	14														
Model	Ours Win (%)	Ours Lose (%)	Tie (%)																												
Alpaca-7B	41	16	6																												
Vicuna-7B	30	21	12																												
ChatGPT	27	22	14																												
May 5, 2023	<p>MosaicML announces MPT-7B, “a commercially-usable, open-source model that matches (and in many ways surpasses) LLaMA-7B.” The context window is 65,000 tokens. (<a href="#">MosaicML, 2023</a>)</p>  <table><tr><th>Model</th><th>Maximum Input Length (tokens)</th><th>Open Source Status</th></tr><tr><td>MPT-7B-StoryWriter-65K+</td><td>65000</td><td>Open Source</td></tr><tr><td>Dolly (Pythia-12B)</td><td>~1000</td><td>Open Source</td></tr><tr><td>GPT-J</td><td>~1000</td><td>Open Source</td></tr><tr><td>GPT-Neox-20B</td><td>~1000</td><td>Open Source</td></tr><tr><td>Llama-7B</td><td>~1000</td><td>Open Source</td></tr><tr><td>Cerebras-GPT-1.3B</td><td>~1000</td><td>Open Source</td></tr><tr><td>StableLM-7B</td><td>~1000</td><td>Open Source</td></tr><tr><td>GPT4</td><td>~1000</td><td>Not Open Source</td></tr><tr><td>GPT4-32K</td><td>~32000</td><td>Not Open Source</td></tr></table>	Model	Maximum Input Length (tokens)	Open Source Status	MPT-7B-StoryWriter-65K+	65000	Open Source	Dolly (Pythia-12B)	~1000	Open Source	GPT-J	~1000	Open Source	GPT-Neox-20B	~1000	Open Source	Llama-7B	~1000	Open Source	Cerebras-GPT-1.3B	~1000	Open Source	StableLM-7B	~1000	Open Source	GPT4	~1000	Not Open Source	GPT4-32K	~32000	Not Open Source
Model	Maximum Input Length (tokens)	Open Source Status																													
MPT-7B-StoryWriter-65K+	65000	Open Source																													
Dolly (Pythia-12B)	~1000	Open Source																													
GPT-J	~1000	Open Source																													
GPT-Neox-20B	~1000	Open Source																													
Llama-7B	~1000	Open Source																													
Cerebras-GPT-1.3B	~1000	Open Source																													
StableLM-7B	~1000	Open Source																													
GPT4	~1000	Not Open Source																													
GPT4-32K	~32000	Not Open Source																													
May 11, 2023	<p>The context window of Claude is expanded to 100,000 tokens, roughly 75,000 words (<a href="#">Anthropic, 2023</a>)</p>																														
May 19, 2023	<p>BLOOMChat, a 176B parameter chatbot, open-sourced by SambaNova achieved a 45% win-rate against GPT-4 across 6 languages in a human preference study. (<a href="#">SambaNova, 2023</a>)</p>																														

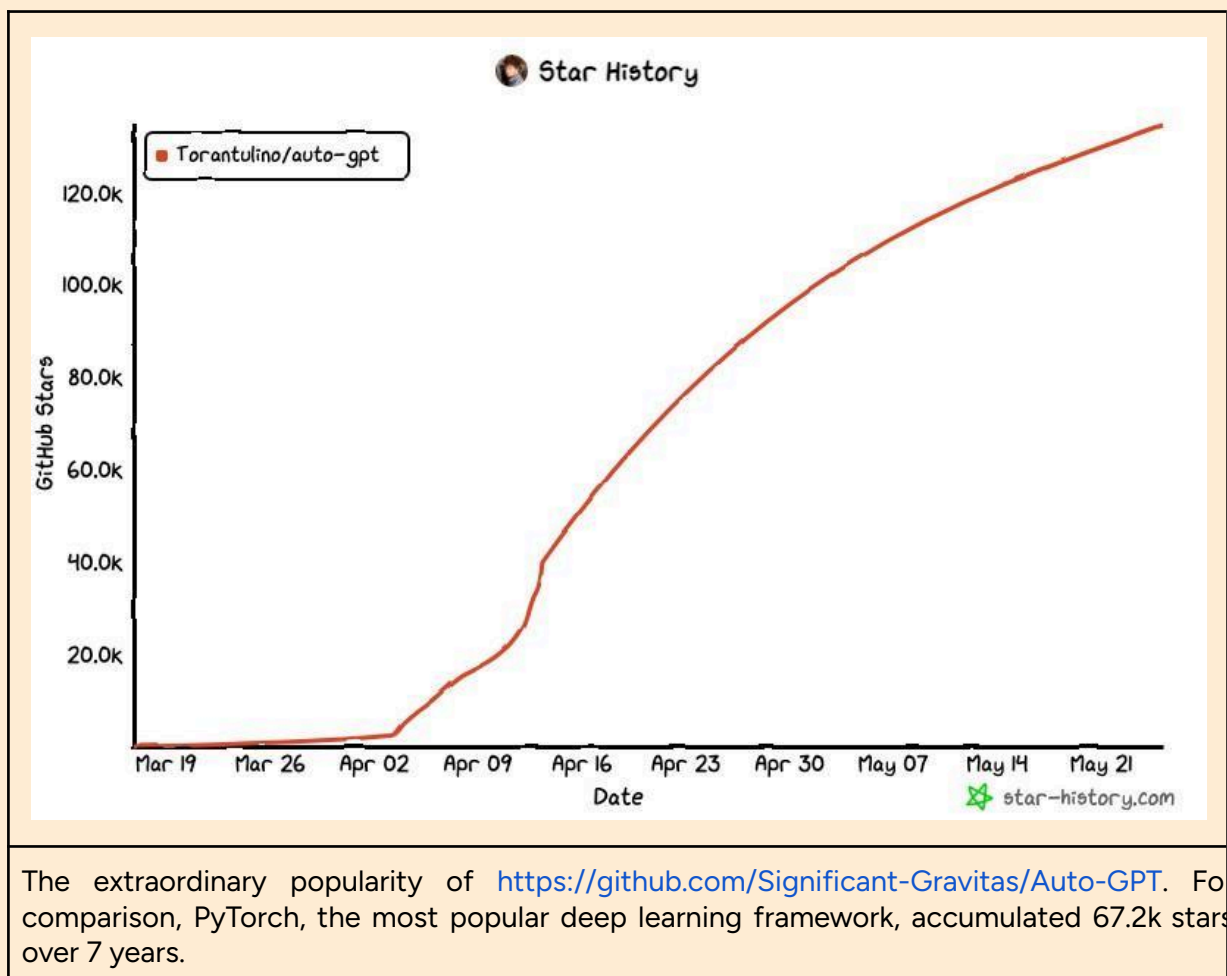
For overall proliferation of LLMs, we reproduce Fig. 1 of Yang et al. ([2023](#)). While this Figure doesn’t allow for comparison of capabilities, we can see about a 2 year gap between GPT-3 and OPT, YaLM and BLOOM. (And a similar 2 year gap between GPT-2 and GPT-J — while GPT-2 was open-sourced, it’s still worthwhile to observe how fast the open-source community caught-up.)\



## Break in the trend?

In recent years, the gap between when capabilities are announced by leading labs and the time they are widely available decreased from ~2 years to ~4 months. We also saw how eager the open-source community is to improve the models they get access to. Newer ideas, such as image-to-image diffusion, prompt generation, and negative prompts were adopted quickly.

This is of course all enabled by industrial labs contributing powerful base models, overcoming the ~\$10m fixed cost in compute alone. Most notably, all the LLMs developed by Meta have been open-sourced to some degree ([Yang et al., 2023](#)).





## State of protein generation<sup>11</sup>

### Natural language and protein language

Ofer et al. ([2021](#)) draw several parallels between natural and protein “languages”:

Like human language, protein sequences can be naturally represented as strings of letters. The protein alphabet consists of 20 common amino acids (AAs) (excluding unconventional and rare amino acids). Furthermore, like natural language, naturally evolved proteins are typically composed of reused modular elements exhibiting slight variations that can be rearranged and assembled in a hierarchical fashion. By this analogy, common protein motifs and domains, which are the basic functional building blocks of proteins, are akin to words, phrases and sentences in human language.

Another central feature shared by proteins and human language is information completeness. Even though a protein is much more than a mere sequence of amino acids – it is also a three-dimensional machine with a determined structure and function – these other aspects are all predetermined by its amino-acid sequence.

Given this, we should not be surprised that NLP methods have been long used for protein modelling. Likewise, it is not surprising that progress in large language models affected protein research.

Indeed, Bommasani et al. ([2021](#)) were eager to name drug discovery as an area awaiting active application: “Foundation models’ generativity can improve the search space and efficiency (see §2.4: reasoning), which not only reduces the amount of experiments but also helps to discover new and better drugs.” Further, they cite protein modeling as “one area where foundation models have shown significant potential for impacting therapeutic design... applications range from predicting viral mutations that can escape a vaccine-induced immune response to predicting protein docking potential for better design of therapeutic antibodies.”

Ferruz & Höcker ([2022](#)) outline six applications where modern NLP tricks could be borrowed for the benefit of protein research, to:

- “(1) generate sequences in unobserved regions of protein space;
- (2) fine-tune sequences of natural protein families to extend their repertoires;
- (3) utilize their encoded vector representations as input for other downstream models for protein engineering tasks;
- (4) generate conditional sequences with specific functional properties;
- (5) design completely novel and purpose-driven receptors and enzymes using encoder-decoder Transformers; and
- (6) gain a more complete understanding of sequence-structure-function relationships and the rules that govern protein folding by interpreting these language models.”

---

<sup>11</sup>We found Peldom ([2021](#)) most useful in writing this review.

## Review of pre-trained protein “language” models<sup>12</sup>

While various language models have been applied to protein sequences, e.g., Yu et al. (2019) used n-gram model, Alley et al. (2019) used LSTM, and Heinzeinger et al., (2019) used ELMo-based approach. We will focus on Transformers pre-trained on protein data as Transformers proved themselves in NLP.

Facebook’s Evolutionary Scale Model (ESM-700M) was the first Transformer model pre-trained on 250 million sequences (86 billion amino acids) of the UniParc database (Rives et al., 2021). Another early model was ProtTrans (Elnaggar et al., 2021), an adaptation of six popular Transformer-based models<sup>13</sup>, released to the community (at <https://github.com/agemagician/ProtTrans>) after being trained on 2.75 billion sequences (390 billion amino acids) taken from UniParc and the ‘Big Fantastic Database’. Both of these showed success in learning “protein grammar” *without* any evolutionary or structural prior information.

The ESM model was later extended to ESM-MSA-1B to make use of Multiple Sequence Alignment information (Rao et al., 2021). Other extensions include ProteinBERT (Ofer et al., 2021), pre-trained on protein sequences and Gene Ontology (GO) annotations; Zhang et al., (2022) considers GO as a factual knowledge graph; Ingraham et al. (2019) augment Transformer with graph-based representations of 3D molecular structure, “leveraging a well-evidenced finding in protein science, namely that long-range dependencies in sequence are generally short-range in 3D space”; Mansoor et al. (2021) encode both sequential and structural information through joint semi-supervised training; Bepler & Berger (2021) also used structural supervision; and Chen et al., (2022) correlated the embeddings learned from sequences and structure by “pseudo bi-level” optimization.

With rapid progress in the medical domain, culminating in Med-PALM-2 (Singhal, 2023) scoring 86.5% on the MedQA benchmark and its answers “being preferred over physician answers by a panel of physicians across eight of nine axes.” It is now natural to incorporate knowledge already created by humans<sup>14</sup> into multimodal models previously limited to using structural information. Towards that goal, Liu et al. (2023) introduced ProteinDT, a multimodal framework that leverages textual descriptions for protein design, and constructed SwissProtCLAP, a large dataset with 441K text and protein pairs extracted from UniProt (UniProt, 2021). This is the first work demonstrating text-to-protein generation. Likewise, Xu et al. (2023) introduced ProtST, a model augmented with biomedical texts, and ProtDescribe, an accompanying dataset.

Following this was the Conditional Transformer Language (CTRL) (Keskar et al., 2019), an autoregressive model that includes conditional tags allowing for controllable generation of

<sup>12</sup> We decided not to review protein generation tools inspired by GANs, VAEs, and Normalizing Flows as these methods have seen less progress compared to LLMs. Some notable/recent models include: ProteinGAN (Repecka et al., 2021), ProteoGAN (Kucera et al., 2022), MSA VAE (Hawkins-Hooker et al., 2021) and ProteinVAE (Lyu et al., 2023). Likewise, we don’t focus on reinforcement learning techniques like once discussed by Mouchlis et al. (2021). Neither we review diffusion models applied to protein generation such as RoseTTAFold Diffusion (Watson et al., 2022) and Chroma (Ingraham et al., 2022); for a review of diffusion models in bioinformatics and protein generation in particular see Guo et al. (2023).

<sup>13</sup> Transformer-XL, BERT, ALBERT, XLNet, T5, and ELECTRA.

<sup>14</sup> According to Abdill (2019), cumulative papers in bioRxiv increased by a factor of 8 from the start of 2016 to the end of 2018, doubling every year.

text<sup>15</sup>. ProGen ([Madani et al., 2020](#)) adapted CTRL for protein use, using UniparKB Keywords as the conditional tags and training on 281 million protein sequences. Later, Madani et al. ([2021](#)) experimentally evaluate model-generated artificial proteins, confirming their ability to perform *de novo* protein generation.

Ferruz et al. ([2022](#)) combined two models: ProtGPT2 ([Ferruz, 2022](#)), used to generate sequences, with ProtT5 ([Elnaggar et al., 2021](#)), used to annotate their functions and discriminate them by desired function. Fine-tuning on protein examples with desired characteristics is used to steer further generations.

Language models can solve a variety of tasks by prefixing the generation with a manually created prompt<sup>16</sup>. Lester et al. ([2021](#)) developed prompt tuning, a method to automatically learn “prompts” in the embedding space. Hesslow et al. ([2022](#)) successfully used it to generate proteins from a target family.

SM-2 ([Lin et al., 2022](#)) is the largest protein LM to date: up to 15 billion parameters. “ESM-2 outperforms all tested single-sequence protein language models across a range of structure prediction tasks.” While ESM-2 is not itself generative, Emami et al. ([2023](#)) introduced an MCMC sampler allowing protein language models to efficiently discover variants with high evolutionary likelihood. And Zheng et al. ([2023](#)) show how to “reprogram sequence-based protein language models [...] to acquire an immediate capability to design preferable protein sequences for given folds.”

Continuing this line of work at Meta, Hie et al. ([2022](#)) leveraged ESM-2 (and an end-to-end protein folding model, ESM Fold ([Lin et al., 2022](#))) to design a modular programming language allowing specification of the desired properties for proteins.

Vu et al. ([2023](#)) point out that, while techniques directly borrowed from NLP work well, we should not expect them to be *optimal* for protein modeling. It is well known that Transformers that use byte pair encoding (BPE) ([Sennrich et al., 2016](#)) to solve out-of-vocabulary problems struggle with character-level problems like arithmetic or anagrams because (some speculate) BPE is not a “natural” fit for the tasks ([Gwern, 2020](#)).

Similarly, we shouldn’t be surprised if BPE or other standard encoders do not immediately align with the most meaningful protein representations. In that direction, Elnaggar et al. ([2023](#)) pursued a biology-informed approach and “through over twenty experiments ranging from masking, architecture, and pre-training data”, derived insights allowing them to train Ankh, a protein language model that surpassed the state-of-the-art performance of ESM-2-15B with fewer parameters (<10% for pre-training, <7% for inference, and <30% for the embedding dimension).

Ferruz & Höcker ([2022](#)) provide a helpful overview of protein LMs up to mid-2022; notable papers published since this are covered in the above.

---

<sup>15</sup>Imagine annotating your text with tags about sentiment, then you can steer the sentiment of text you generate by prompting the model using an appropriate tag.

<sup>16</sup>Famously, GPT-2 learned to summarize a text if you prompt it like “<Text> TLDR:”.

Table 2: List of representative pLMs

Model and Repository	Approach	Input	Network	#Embedding	#Param.	Pre-training Database
NetSurfP-2.0 (Klausen et al., 2018)	Supervised	MSA, Structure	CNN, BiLSTM	2048	N/A	PDB, UniRef30
SPIDER3-Single (Heffernan et al., 2018)	Supervised	Sequence, Structure	LSTM-BRNN (Heffernan et al., 2017)	1024, 512	N/A	12442 proteins
SeqVec (Heinzinger et al., 2019)	Unsupervised	Sequence	ELMo	1024	~93.6M	UniRef50
UniRep (Alley et al., 2019)	Unsupervised	Sequence	mLSTM (Krause et al., 2016)	1900	~18.2M	UniRef50
SSA (Bepler and Berger, 2019)	Supervised	Sequence, Structure	BiLSTM	100, 512	N/A	Pfam, SCOP
DeepPrime2Sec (Asgari et al., 2019)	Supervised	MSA, Structure	ELMo, CNN, BiLSTM	N/A	N/A	UniRef50, Swiss-Prot, CullPDB
Ingraham et al. (2019)	Unsupervised	Structure	Transformer	128	N/A	CATH4.2
TAPE (Rao et al., 2019)	Unsupervised	Sequence	LSTM	2048	N/A	Pfam
ESM-1b (Rives et al., 2019)	Unsupervised	Sequence	Transformer	768	38M	
UDSMProt (Strodthoff et al., 2020)	Unsupervised	Sequence	Transformer	1280	650M	UniParc
CPCProt <sub>GRU</sub> (Lu et al., 2020)	Unsupervised	Sequence	LSTM	400	~24M	Swiss-Prot
CPCProt <sub>LSTM</sub> (Lu et al., 2020)	Unsupervised	Sequence	GRU (Cho et al., 2014)	1024	8.4M	Pfam
Sturmfels et al. (2020)	Unsupervised	Sequence	LSTM	2048	71M	Pfam
ProGen (Madani et al., 2020)	Supervised	MSA	Transformer	N/A	N/A	Pfam
ProTGen (Madani et al., 2020)	Unsupervised	Sequence, Property	Transformer	1028	1.2B	~280M proteins
ProTXL (Elnaggar et al., 2021)	Unsupervised	Sequence	Transformer-XL	1024	562M	BFD100, UniRef100
ProTERT (Elnaggar et al., 2021)	Unsupervised	Sequence	BERT	1024	420M	BFD100, UniRef100
ProXLNet (Elnaggar et al., 2021)	Unsupervised	Sequence	XLNet	1024	409M	UniRef100
ProTAlb (Elnaggar et al., 2021)	Unsupervised	Sequence	ALBERT	4096	224M	UniRef100
ProTElectra (Elnaggar et al., 2021)	Unsupervised	Sequence	ELECTRA	1024	420M	UniRef100
ProT5 (Elnaggar et al., 2021)	Unsupervised	Sequence	T5	1024	11B	UniRef50, BFD100
SPOT-iD-Single (Singh et al., 2021a)	Supervised	Sequence, Structure	BiLSTM, ResNet (He et al., 2016)	256	N/A	39120 proteins
ProSE (Bepler and Berger, 2021)	Supervised	Sequence, Structure	BiLSTM	1024	1M	UniRef90, SCOPe
MSA Transformer (Rao et al., 2021a)	Unsupervised	MSA	Transformer	768	100M	UniRef50, UniClust30
ESM-1v (Meier et al., 2021)	Unsupervised	Sequence	ESM-1b	1280	650M	UniRef90
ESM-1F1 (Hsu et al., 2022)	Supervised	Sequence, Structure	GVP (Jing et al., 2020), Transformer	512	142M	UniRef50, CATH
ProteinBERT (Ofer et al., 2021c)	Supervised	Sequence	Transformer	128, 512	~16M	UniRef90
Fold2Seq (Cao et al., 2021)	Supervised	GO annotation	Transformer			
AminoBERT (Chowdhury et al., 2022)	Supervised	Sequence, Structure	Transformer	256	N/A	CATH4.2
Evoformer (Jumper et al., 2021)	Unsupervised	Sequence	Transformer	3072	N/A	UniParc
OmegaPLM (Wu et al., 2022b)	Supervised	MSA, Structure	Attention network	384, 128	93M	PDB, BFD, UniClust30, etc.
OntoProtein (Zhang et al., 2022b)	Unsupervised	Sequence	GAU (Hua et al., 2022)	1280	670M	UniRef50
PeTriBERT (Dumortier et al., 2022)	Supervised	Sequence, GO	ProtBERT, BERT	1024	N/A	ProteinKG25
MSA2Prot (Bepler and Ram, 2022)	Unsupervised	Sequence, Structure	BERT	3072	<40M	AlphaFoldDB
PMLM (He et al., 2022)	Unsupervised	MSA	Transformer	768	N/A	Pfam
ProGen2 (Nijkamp et al., 2022)	Unsupervised	Sequence	Transformer	1280	715M	UniRef50
Tranception (Notin et al., 2022)	Unsupervised	Sequence	Transformer	4096	6.4B	UniRef90, BFD30
ProtGPT2 (Ferruz et al., 2022)	Unsupervised	Sequence	Transformer	1280	700M	UniRef100
RITA (Hesslow et al., 2022)	Unsupervised	Sequence	GPT-2	1280	738M	UniRef50
ESM-2 (Lin et al., 2022)	Unsupervised	Sequence	GPT-3	2048	1.2B	UniRef100
	Unsupervised	Sequence	Transformer	5120	15B	UniRef50

All examples report the largest model of their public series, the model name with colour is linked with GitHub or server page. Approach and database are listed for the pre-training stage, and the latter is elaborated on in Section 8. Input is classified into protein sequence, MSA, structure (structural features or coordinates), and function. Network displays high-level backbone models preferentially if they are used. #Embedding means the dimension of embeddings; #Param., the number of parameters of network; M, millions; B, billions; T, trillions; N/A, null; &, and; <, less than; ~ estimated data.

(Ferruz & Höcker, 2022)

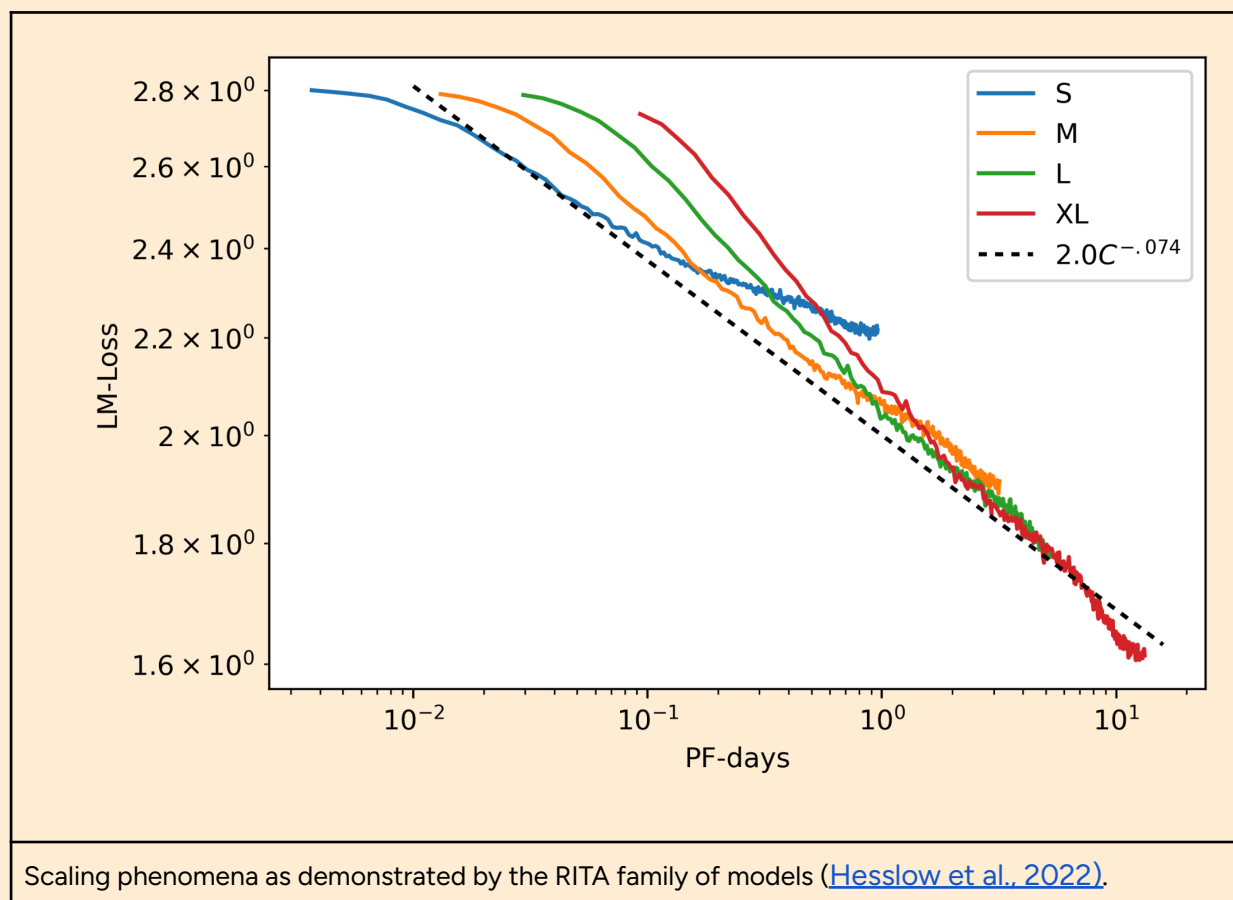
## Scaling laws for protein generation

With the rapid growth of protein language model scales, from 38M parameters in July 2019 (Rao et al., 2019) to 15B parameters in July 2022 (Lin et al., 2022), it is natural to ask what the role of scaling (see [Scaling Laws](#)) in improving performance of generative models will be.

Hesslow et al. (2022) conducted the first and only scaling study, showing how capabilities evolve with increased model size. RITA is a family of generative models with parameter counts scaled between 85M, to 300M, to 680M, to 1.2B. This increase in size leads to a proportional performance improvement: they found “an exponent of 0.074, significantly steeper than one observed in NLP” (compared with an exponent of 0.05 found by Kaplan et al. (2020) for human text<sup>17</sup>).

<sup>17</sup>It's worth noting that Hoffmann et al. (2020) throughout analysis also suggest steeper scaling than suggested by Kaplan et al. (2020).

Further they note that “all but our smallest model still appear to be undertrained”, despite being trained for 150 billion amino acids” — a finding which mimics Kaplan et al. (2020)’s observation that optimal training should stop *before* convergence.



A similar trend was reinforced by ProGen2 (Nijkamp et al., 2022), a family of models trained on different sequence datasets and whose parameter count was scaled up to 6.4B (Elnaggar et al., 2023).

Given that this is early work, we should not be surprised if this estimated exponent changes, making compute-efficient training look different, as happened with Chinchilla (Hoffmann et al., 2020) for human text. On present evidence, we agree with Elnaggar et al. (2023) “that scaling-up protein language models may be significantly more impactful than scaling-up natural language models.”

Knowing the functional form of the loss  $L(N, D)$  would be particularly useful in guiding further scaling, especially in the regime where data is a potential bottleneck.

$$L(N, D) = \underbrace{\frac{A}{N^\alpha}}_{\text{finite model}} + \underbrace{\frac{B}{D^\beta}}_{\text{finite data}} + \underbrace{E}_{\text{irreducible}}$$



We should thus expect *more* improvement from the next generation of compute-optimal models.

## Challenges<sup>18</sup>

After observing the effectiveness of techniques borrowed from NLP for protein design and the apparently huge potential of scaling up protein language models, we are left wondering whether training a 175 billion parameter model and incorporating fine-tuning could lead to a revolution in protein generation similar to the ChatGPT breakthrough.

It's unclear. While we noted domain similarities that seem to make NLP techniques effective at protein generation, we need to also note the challenges and bottlenecks raised throughout literature.

We now review dissimilarities between NLP and “protein language processing” (PLP) through the lens of the [AI Triad](#): data, algorithms, and compute.

## Compute

On a first look, compute is just another commodity which applies roughly similarly to any input domain. Given that similar architectures are used both in NLP and PLP, namely Transformers, again *prima facie* we might not expect much difference in training cost or effectiveness.

Nonetheless, there might be differences in willingness to use compute, to spend whole research budgets on large single experiments. We previously noted the model size hike from 38M parameters in July 2019 ([Rao et al., 2019](#)) to 15B parameters in July 2022 ([Lin et al., 2022](#)), yet to be surpassed. We now compare trends between protein LMs and text LMs.’

### Trend in parameter counts

Building on Romero-Romero et al. ([2023](#)), we collect model sizes of protein-LMs below. Record-setting systems are **bolded**. Final-run training compute is not reported, so we estimate it with the methods in Sevilla et al. ([2022](#)).<sup>19</sup> These are uncertain and may be off by a factor of 5, owing (among other things) to large unreported differences in GPU utilization.

---

<sup>18</sup> Some readers might be interested in Akbar et al's ([2021](#)) review of challenges to machine-learning based design of fit-for-purpose monoclonal antibodies.

<sup>19</sup>For reference, the text-LM GPT-3 took 3.14e23 FLOPs for its largest training run ([Brown et al 2020](#)).

Release	System	Max Params	Estimated FLOPs	Citation	Institution
01/2023	Ankh	1.15B	2.1e19 <sup>20</sup>	<a href="#">Elnaggar et al. 2023</a>	Proteinea
12/2022	ZymCTRL	762M	—	<a href="#">Munsamy et al 2022</a>	Basecamp
<b>10/2022</b>	<b>ESM-2</b>	<b>15B</b>	<b>7.8e22<sup>21</sup></b>	<a href="#">Lin et al. 2023</a>	<b>Meta AI</b>
06/2022	ProGEN2	6.4B	1.3e22 <sup>22</sup>	<a href="#">Nijkamp et al. 2022</a>	Salesforce
05/2022	DistilProtBert	230M	—	<a href="#">Geffen et al. 2022</a>	
05/2022	RITA	1.2B	4.1e20 <sup>23</sup>	<a href="#">Hesslow et al. 2022</a>	LightOn
05/2022	Tranception	700M	—	<a href="#">Notin et al. 2022</a>	
05/2022	ProtGPT2	762M	—	<a href="#">Ferruz et al. 2022</a>	
01/2022	DARK	110M	—	<a href="#">Moffat et al. 2022</a>	
08/2021	ProteinLM	3B	3.9e21 <sup>24</sup>	<a href="#">Xiao et al. 2021</a>	Beijing Academy of AI
05/2021	ProteinBERT	16M	—	<a href="#">Brandes et al. 2022</a>	
12/2020	ESM-1	670M	—	<a href="#">Rives et al. 2021</a>	Meta AI
09/2020	PRoBERTa	44M	—	<a href="#">Nambiar et al. 2020</a>	
<b>07/2020</b>	<b>ProtTrans</b>	<b>11B</b>	Uncertain <sup>25</sup>	<a href="#">Elnaggar et al 2021</a>	<b>TUM, ORNL</b>
<b>03/2020</b>	<b>ProGEN</b>	<b>1.2B</b>	—	<a href="#">Madani et al. 2020</a>	<b>Salesforce</b>
<b>06/2019</b>	<b>TAPE</b>	<b>38M</b>	—	<a href="#">Rao et al. 2019</a>	

<sup>20</sup>Ankh\_large: 68 epochs x 45M proteins x 1.15 B "connections" x 6 (per [C ≈ 6ND](#)).

<sup>21</sup>ESM-2-15B: 270000 updates x 3.2M batch size x 15 B "connections" x 6. Alternatively, 60 days x 512 V100s x an imputed 30% utilization.

<sup>22</sup>PROGEN2-xlarge: Uses UniRef90 (144 million proteins) and BFD30 (48 million proteins). 350,000 steps x 1m batch size x 6.4 B "connections" x 6.

<sup>23</sup>RITA-XLARGE: 280m proteins. 25000 V100-hours x an imputed 30% utilization.

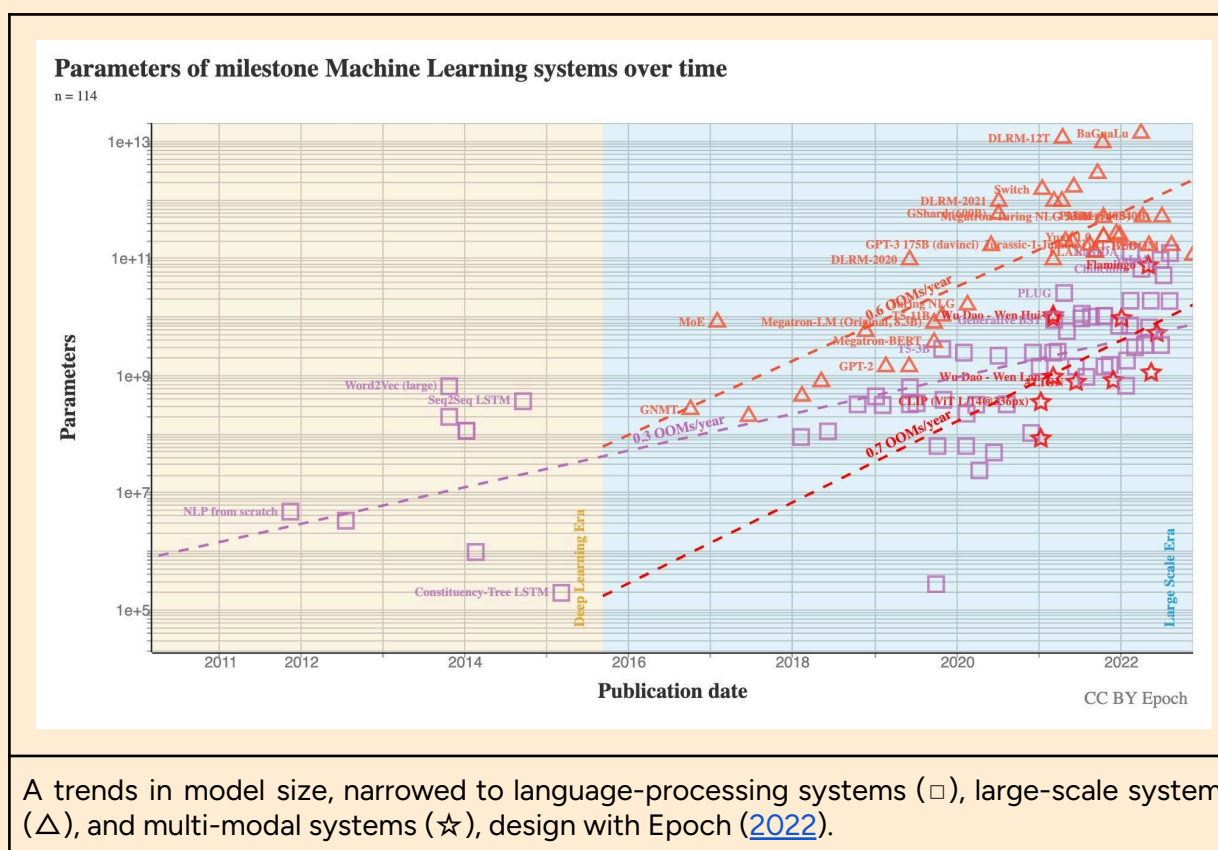
<sup>24</sup>ProteinLM-3B: 21 days x 480 V100-32GBs x an imputed 30% utilization. May be overestimated by an extra factor of 2, given that they seem to have trained the 1.2B and the 3B at the same time.

<sup>25</sup>ProtT5-XXL-U50: ( (920k + 343k) updates / (2122m proteins / 44000? batch size) x 2122m proteins x 11B

"connections" x 6 = 3.4e23. However, batch size is unreported for this model size.

(20? hours per epoch \* 70? epochs) x 5616 V100s x Half precision x an imputed 30% utilization = 2.4e23. Epochs and GPU-hours unreported for this model size.

It seems useful to compare with how parameter-count increased in NLP:



We see that protein LMs are notably smaller than LLMs, but have scaled remarkably fast.

## Future willingness to spend

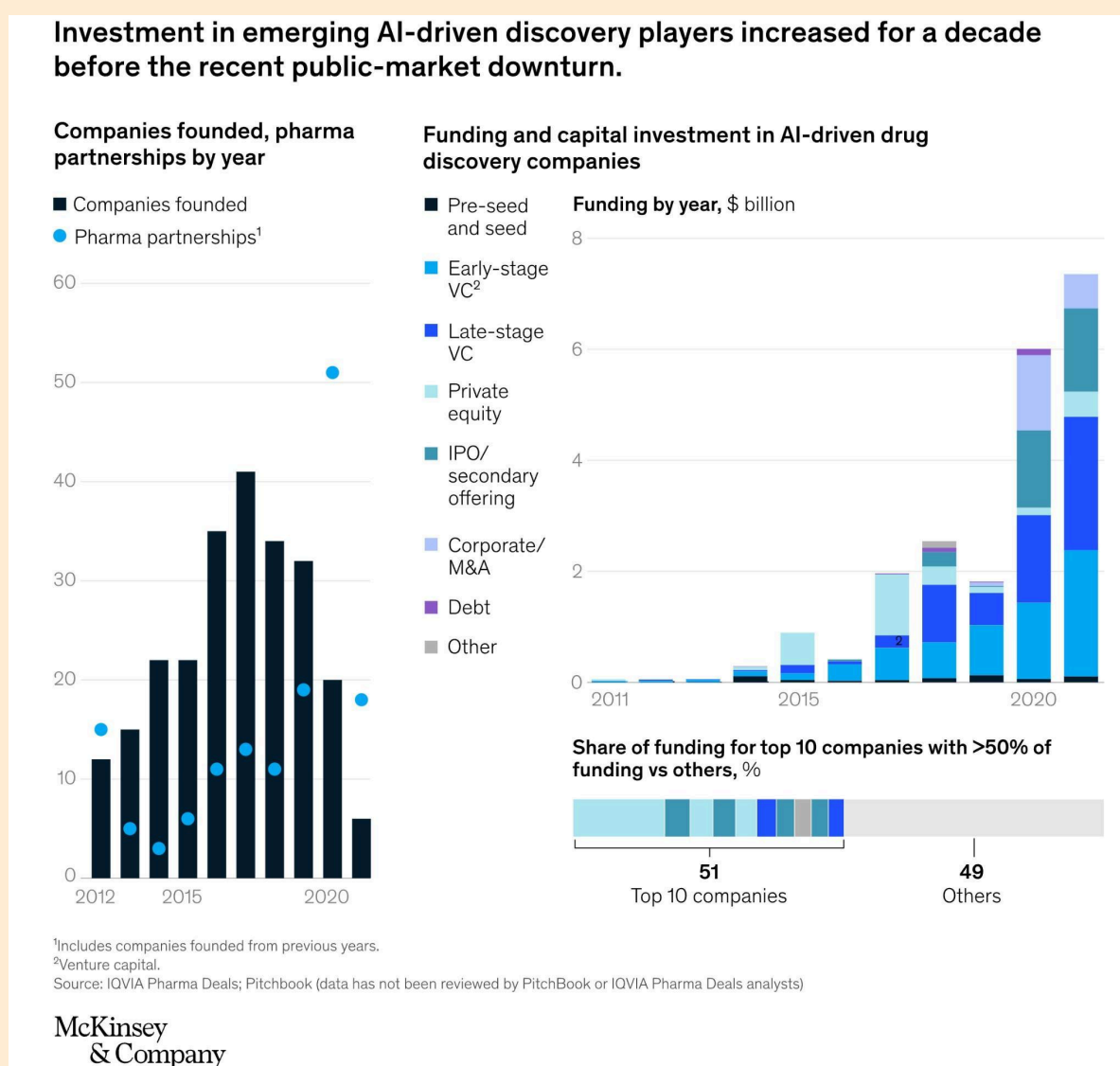
Training GPT-3-sized models is both expensive (with a cost exceeding \$5M for one full training run) and technically challenging – only a few teams have the technical expertise and financial means to scale training up to thousands of GPUs and to sustain one such computation over a month.

We can see that Salesforce Research contributed the first really large protein LM, and they continue to be active in the field ([Salesforce, 2023](#)). The present leader is Meta AI, which has<sup>26</sup> a dedicated *Meta Fundamental AI Research Protein Team* ([Meta, 2023](#)). Likewise, Google DeepMind, while currently absent from our specific protein-LM space, contributed AlphaFold and is generally active in “AI for science” ([DeepMind, 2022](#)); and Isomorphic Labs is a drug-discovery company that was established as a spin-off from Deepmind. Recently, Microsoft Research announced a AI4Science, a new initiative to accelerate progress in science ([Bishop, 2022](#)). While significantly fewer people work on protein-LMs than on LLMs at these labs, we should nonetheless anticipate future scaling and for advances in NLP to continue to percolate into protein design.

<sup>26</sup>After completing this report, we learned that this team was recently laid off in full.

Another contender are pharmaceutical companies, presently missing from the list of contributors, but who have strong financial incentive to develop better models. Pharma companies have huge R&D expenses, with a Morgan Stanley (2022) estimate finding that “The median investment required to bring a new drug to market is estimated to be nearly \$1 billion, while the true cost of research and development may be as high as \$2.5 billion per marketed therapy, when factoring in abandoned trials and clinical failures” and estimate that “a 20% to 40% reduction in costs for preclinical development across a subset of U.S. biotech companies could generate the cost savings needed to fund the successful development of four to eight novel molecules” — a significant incentive.

Indeed, investment in drug discovery companies is increasing rapidly according to Devereson et al (2022):



For a forward-looking prognosis, Data Bridge Market Research (2022) predicts that the AI bioinformatics market is expected to reach \$37B by 2029, growing 42.7% per year.

## Data<sup>27</sup>

Data is the part of the AI triad with the largest discrepancy between protein and text. Data is expensive to collect and exists in a limited amount – the UniRef50 dataset is seemingly the only choice for large scale pre-training at present.

By contrast, data in the English language is abundant and diverse ([Villalobos et al., 2022](#)). Datasets such as The Pile ([Gao et al., 2021](#)), MassiveText ([Rae et al., 2021](#)), and the PaLM pre-training dataset ([Chowdhery et al., 2022](#)) are 825 GB, 10.5 TB, and 6.7 TB respectively. BigQuery and BigPython are datasets of code reaching 4 TB and 5.5 TB respectively ([Nijkamp, 2022](#)). We reproduce Table 4 from Hu et al. ([2022](#)) that reviews databases available for pre-training:

Dataset	Proteins	Size in GB	Description	Link
UniProtKB/Swiss-Prot	500K	0.59GB	knowledgebase	<a href="https://www.uniprot.org/uniprotkb?query=*">https://www.uniprot.org/uniprotkb?query=*</a>
UniProtKB/TrEMBL	229M	146GB	knowledgebase	<a href="https://www.uniprot.org/uniprotkb?query=*">https://www.uniprot.org/uniprotkb?query=*</a>
UniRef100	314M	76.9GB	clustered sets sequences	<a href="https://www.uniprot.org/uniref?query=*">https://www.uniprot.org/uniref?query=*</a>
UniRef90	150M	34GB	90% identity	<a href="https://www.uniprot.org/uniref?query=*">https://www.uniprot.org/uniref?query=*</a>
UniRef50	53M	10.3GB	50% identity	<a href="https://www.uniprot.org/uniref?query=*">https://www.uniprot.org/uniref?query=*</a>
UniParc	528M	106GB	Sequence	<a href="https://www.uniprot.org/uniparc?query=*">https://www.uniprot.org/uniparc?query=*</a>
PDB	190K	50GB	3D structure	<a href="https://www.wwpdb.org/ftp/pdb-ftp-sites">https://www.wwpdb.org/ftp/pdb-ftp-sites</a>
CATH4.3	N/A	1073MB	hierarchical classification	<a href="https://www.cathdb.info/">https://www.cathdb.info/</a>
BFD	2500M	272GB	sequence profile	<a href="https://bfd.mmseqs.com/">https://bfd.mmseqs.com/</a>
Pfam	47M	14.1GB	protein families	<a href="https://www.ebi.ac.uk/interpro/entry/pfam/">https://www.ebi.ac.uk/interpro/entry/pfam/</a>
AlphaFoldDB	214M	23TB	predicted structures	<a href="https://alphafold.ebi.ac.uk/">https://alphafold.ebi.ac.uk/</a>

<sup>27</sup>See also an editorial in Nature ([2023](#)) pointing out data as a bottleneck for applicability of AI to chemistry.



ProteinKG25	5.6M	147MB	a KG dataset with GO	<a href="https://drive.google.com/file/d/1iTC2-zbvYZCDhWM_wxRufCvV6vvPk8HR">https://drive.google.com/file/d/1iTC2-zbvYZCDhWM_wxRufCvV6vvPk8HR</a>
Uniclust30	N/A	6.6GB	clustered protein sequences	<a href="https://uniclust.mmseqs.com/">https://uniclust.mmseqs.com/</a>
SCOP	N/A	N/A	structural classification	<a href="http://scop.mrc-lmb.cam.ac.uk/">http://scop.mrc-lmb.cam.ac.uk/</a>
SCOPe	N/A	86MB	extended version of SCOP	<a href="http://scop.berkeley.edu">http://scop.berkeley.edu</a>

With the exception of 3D structures generated by AlphaFold, we see that the total size is ~720 GB. This is about an order of magnitude less than the datasets used to train frontier text models, even the smaller ones (for example, LLaMa ([Touvron, 2023](#)) was trained on [1–1.4B tokens](#), roughly 3.5–5 TB<sup>28</sup>).

Still, in some domains less (higher quality) data is enough for a highly capable model. CodeGen ([Nijkamp et al., 2022](#)) filters BigQuery and BigPython down to 350 GB and 220 GB respectively. But their follow-up work CodeGen2 ([Nijkamp et al., 2023](#)) uses Stack ([Kocetkov et al., 2022](#)), (3 TB) and the full BigPython (5.5 TB).

Presently, larger protein models mostly employ UniRef50, which is a mere 10GB, so there is some room to scale up models using the above sources. But issues with the data quality in the life sciences may not allow that.

### Issues with data quality

High-throughput molecular assays and the curation of data resources (such as UniProt) are the “engine” of the field ([Ofer et al., 2021](#)). But high-throughput techniques inevitably come with certain quality limitations, according to Schnoes et al. ([2013](#)).

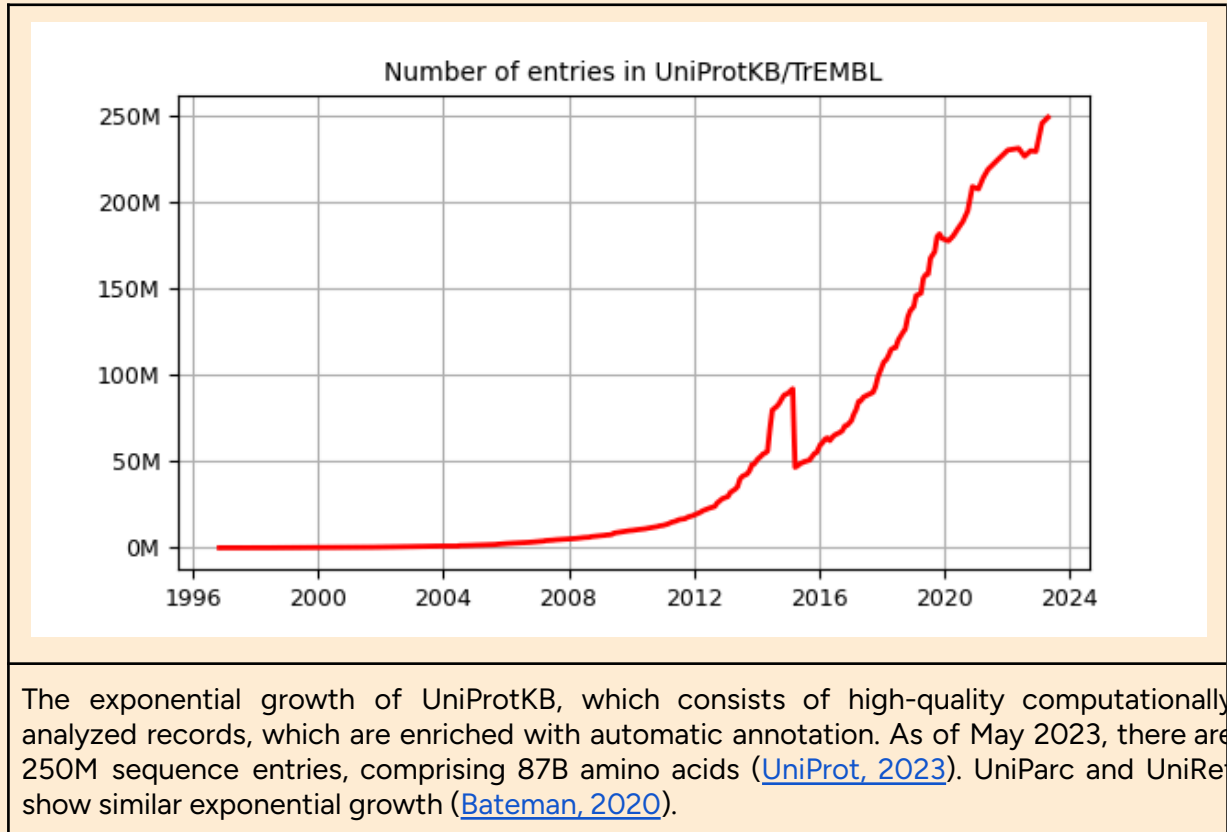
The data quality of some popular protein sequence datasets might be an issue. Ofer et al. ([2021](#)) note that this data is smaller, sparser and more biased, which all cause challenges for pre-training using auxiliary tasks.

In the experiments of Prot-T5, it was found that UniRef50 outperformed larger datasets such as UniRef100 and BFD ([Elnaggar et al., 2020](#)). They trace the high quality of UniRef50 to its lack of duplication and diversity in sequences. Indeed, Singer et al. ([2020](#)) discover “systematic biases have influenced the completeness of data and the patterns of ‘dark matter’ of undiscovered findings.”

<sup>28</sup>According to <https://paperswithcode.com/dataset/massivetext>, 300B tokens are 12.8% of the MassiveText dataset and the whole dataset is ~10.5 TB.

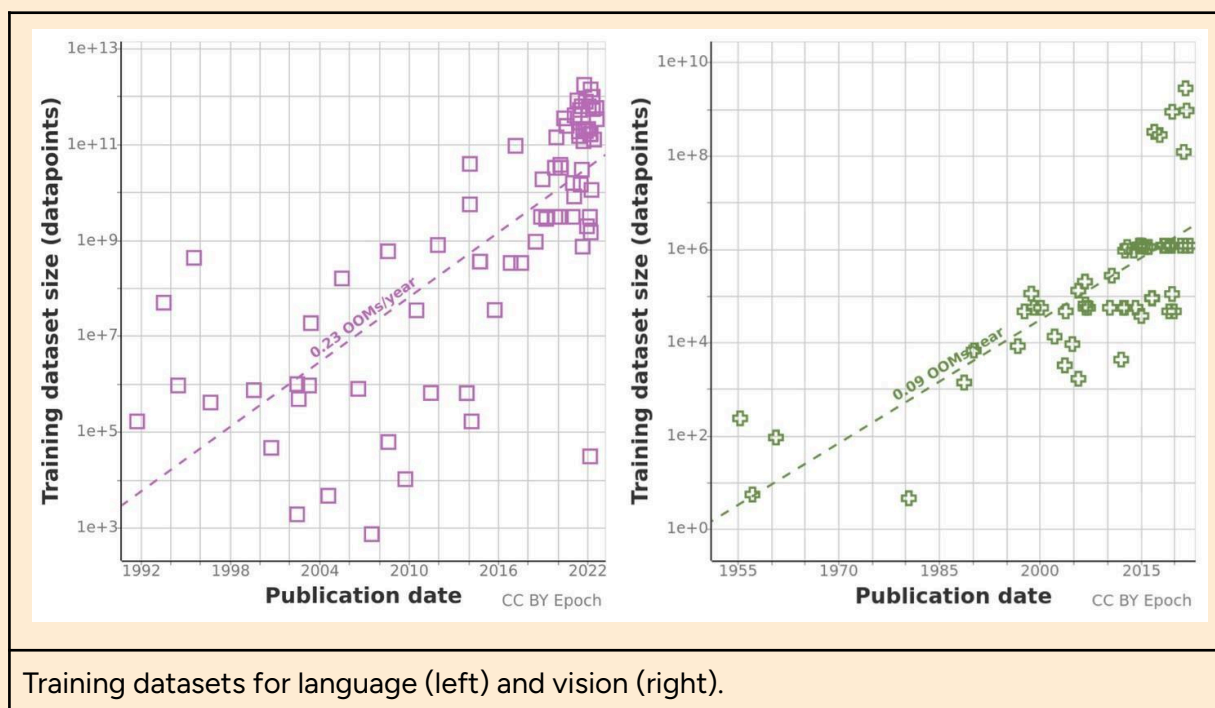
## Growth of existing databases

In this subsection we examine how current databases are growing in size and whether we expect a rapid influx of new protein sequences.



We see UniProtKB doubling in size in the last 7 years, corresponding to 11% annual growth or 0.043 OOMs/year. This is smaller than (but comparable with) historical growth rates of NLP dataset sizes (0.23 OOMs/year) and CV (0.09 OOMs/year) dataset sizes ([Villalobos & Ho, 2022](#)).

But as we see from the following plots, after the transition to the “Foundation Models” paradigm ([Bommasani, 2021](#)) and the “Large-Scale [Deep Learning] Era”, around 2016, ([Sevilla et al., 2022](#)), a notable acceleration of dataset sizes occurred.



Villalobos et al. (2022) note that we are likely to run out of data “between 2030 and 2040 for language models and between 2030 and 2060 for image models. This is particularly true for high-quality language data, which seems likely to be exhausted by 2027”.

We have even greater constraints on high quality protein-related data in the future (if not already), as most models<sup>29</sup> regardless of their size (e.g., Prot-T5 and ESM-2-15B) are trained on UniRef50 alone, which indicates no low-hanging fruit available to academic or industry labs. It remains to be seen if (e.g.) active learning algorithms enable effective learning from larger protein sequence datasets like UniRef100.

<sup>29</sup>Note that ProGen2 used UniRef90 (144 million proteins) and BFD30 (48 million proteins).

## New sources of data?

One way to expand protein sequence data might be to use the 2.4B<sup>30</sup> genomic sequences from Whole Genome Shotgun ([NIH, 2023](#)). Proteins are produced by the translation of messenger RNA; datasets like UniParc already contain protein sequences *predicted* from mRNA readings. Many of the 250M sequences in the NIH's GenBank repository have likely already been added to existing protein datasets, however. While WGS genomic sequences are incomplete, these sequences still contain usable information. We should not forget that WGS data is likely to suffer from the same set of quality issues most notably diversity..

We are aware of some companies (like [Basecamp Research](#)) assembling novel datasets specifically for AI-powered drug-discovery ([Eisenstein, 2023](#)). With growing investment in AI-based drug discovery and general growth in bioinformatics, we should expect more specialist data brokers to supply datasets constructed particularly for these domains.

### Possibility of synthetic data?

AlphaFold2 greatly expanded the world's total store of protein structure data. We can wonder if other advances in AI are likely to greatly expand the data stock. One path for this is generating synthetic data from imperfect deep-learning simulations of proteins.

Synthetic data is often for a variety of machine learning tasks: models for image segmentation are often pre-trained on synthetic data; and the classic LSTM ([Hochreiter & Schmidhuber, 1997](#)) was validated on synthetic data. Sandve & Greiff ([2022](#)) argue that carefully constructed simulated data might in some cases be superior to experimental data as "often available experimental data do not have the size, resolution and sufficient set of controls that would allow for rigorous method assessment." Further, as discussed above, experimental data can be biased because its collection depends on very specific ideas that an experimenter wants to validate and assumptions they operate under, and is as such a highly selected subset of all proteins. Synthetic data might flexibly allow for "sampling" parts of protein space with very different background assumptions and auxiliary hypotheses ([Duhem, 1914](#)).

---

<sup>30</sup> We estimate this translating into 1.6TB of protein sequence data:

- "More than 95% of the protein sequences provided by UniProtKB come from the translations of coding sequences (CDS) submitted to the EMBL-Bank/GenBank/DDBJ nucleotide sequence resources" ([UniProt, 2023](#)). Though, [the most recent \(May 2023\) release](#) notes say that 77% of entries list "Submitted to EMBL/GenBank/DDBJ" among their reference.
  - EMBL-Bank/GenBank/DDBJ share data with each other, so we can think that data come from GenBank.
- UniProtKB consists of UniProtKB/TrEMBL and UniProtKB/Swiss-Prot. As estimated by Hu et al. ([2022](#)), the former has 229M sequences worth 146GB, and the latter has 500K sequences worth 0.59GB. At the current size of 250M sequences, we can estimate the total size of 160GB.
- As of April 2023, GenBank has 243M sequences and 1.8T bases. As any base is one of A, T, G, or C (so 2 bits), we can estimate the size as 1.8T \* 2 bits or 450GB.
  - So .77 to .95 of UniProtKB's 160GB protein sequence data came from GenBank's 450GB of genomic sequence data.
- As of April 2023, Whole Genome Shotgun has 2440M sequences and 20.9T bases, which leads to a size of 5.2TB.
  - So, at the same rate as GenBank, we can expect 1.6TB worth of protein sequences (= 5.2TB / 450 GB \* 160 GB \* (0.77 + 0.95) / 2). We probably should adjust that down as the nature of WGS might lead to lower yields.

To give an example, Robert et al. ([2021](#)) developed the *Absolut!* software suite that allows parameter-based unconstrained generation of synthetic lattice-based 3D-antibody-antigen binding structures. This allows us to translate antibody specificity prediction problems into ML tasks.

Which avenues yield promising synthetic data for protein design? Recent progress in simulating physical phenomena with deep learning ([Bezenac et al., 2018](#); [Noé et al., 2020](#) [Hsin-Yuan Huang et al., 2022](#)) makes us wonder if various protein interactions can be simulated with these AI techniques, yielding intriguing data, or if AI-enabled modeling could make “digital twins” of wet labs much more effective, significantly increasing throughput.

Such simulations take time to construct and validate, but techniques focused on sample-efficient black-box optimization could still make great use of it. One such framework for wet-lab data was developed and tested in Angermueller et al. ([2020](#)).

Could the problem go away?

Sam Altman [has claimed](#) that GPT-4 cost more than \$100M (tacitly, for a single full training run); future systems are likely to cost even more. At this point, it’s already economical for leading labs to intensely research increasing sample efficiency. Another contributing factor is that we will run out of high-quality data (and the most useful data<sup>31</sup> might run out as soon as 2025) ([Villalobos et al., 2022](#)).

As Socher et al ([2022](#)) demonstrate, focusing system training on only the highest quality data can make learning drastically more efficient, leading to more accurate models at a fraction of the compute cost.

## Algorithms

So the state of protein data is fairly gloomy. Improving and scaling algorithms in light of these issues with data also faces serious challenges, but we highlight some opportunities from better algorithms in the following subsection.

How much algorithmic progress should we expect?

Algorithmic progress in computer vision was examined by Hernandez & Brown ([2020](#)), who found a 44-fold reduction in compute required to reach a given level of performance over the 8 years since AlexNet ([Krizhevsky et al., 2012](#)). So algorithmic progress has been a strong contributor to AI progress, outpacing the original Moore's Law rate of improvement in hardware efficiency. Besiroglu & Besiroglu ([2022](#)) examined progress in computer vision more closely and concluded “that algorithmic improvements in image classification have been roughly as important as the scaling of compute for driving performance improvements (with the caveat that algorithmic progress was more crucial in early years, and compute scaling more important in later years).”

---

<sup>31</sup> E.g., code.



While only CV has so far been examined thoroughly, this finding is “consistent with work in other domains, such as computational solvers for linear programs, as well as computer chess, [where software and hardware improvements] are both important drivers of performance improvements, and that hardware improvements generally account for slightly more of total performance improvements” according to Besiroglu & Besiroglu ([2022](#)).

## Talent

It's obvious that the application of LLMs to protein-generation has attracted much less research talent than the application of LLMs to natural language. We list 16 language models in [Trends in parameter-counts](#), while Hannibal046 ([2023](#)) lists 46 milestone LLM papers; maybe 6 of these 16 papers seem similarly epochal to us.

But due to the lag<sup>32</sup> between protein-LM and LLM methodologies, the protein-LM community can borrow advances from the LLM community. And not only advances: as it gets harder and harder to get marginal gains on the LLM frontier, protein generation may “borrow” LLM researchers.

In closing, we can wonder what the protein analogs of in-context learning ([Wei et al., 2022](#)), chain-of-thought prompting ([Wei et al., 2022](#)), and self-consistency through chain-of-thought ([Wang et al., 2022](#)) will be.

## Gains from careful engineering

Due to the stated dissimilarities between the protein and natural language domains, we should not expect the many ad hoc rules of thumb developed over the years in NLP to be optimal for protein language models.

## Protein language

Vu et al. ([2023](#)) outline a linguistically-inspired roadmap for building a biologically reliable protein language. By highlighting how the original linguistic-science intent shaped different tools in NLP, they suggest that a more biologically-informed approach could benefit the systems. Specifically they discuss that:

- pre-training data should reflect the goals of the downstream task;
- tokenization should aim for biologically meaningful units;
- token embeddings should capture protein function;
- and interpretability methods affect learnable patterns.

With this biological approach in mind, Elnaggar et al. ([2023](#)) run more than 20 experiments about different architectural and data choices such as masking, layers, pre-training data. This leads them to Ankh-1.2B, a protein language model that surpassed the state-of-the-art

---

<sup>32</sup>Based on the scale of the models and some key results like scaling, we can guess that protein-LMs are about 2 years behind LLM in terms of adopting frontline innovations and scaling-up.

performance of ESM-2-15B with much fewer parameters (<10% for pre-training, <7% for inference, and <30% for the embedding dimension).

It seems plausible that future algorithmic progress could yield similar gains. Two areas where more progress would be beneficial are multi-modality and effective learning from small datasets.

### Multi-modality and small datasets

In [Review of pre-trained protein “language” models](#), we separately highlighted attempts to use additional information – be it protein structure<sup>33</sup>, homologies<sup>34</sup>, protein annotation or even bio-medical texts. Further, we can integrate more biochemical data and even aim for a unified model of drug discovery – using different input modalities in one system has often been beneficial, even if at first sight they seem too dissimilar. Incorporating data across biochemical domains should lead to disproportionate gains ([Rao et al., 2019](#), [Buehler, 2023](#)).

A particular challenge here is data availability: for many modalities the existing data is sparse. ProteinDT authors ([Liu et al., 2023](#)) note that “[Protein DT] has 441K data pairs, yet such a dataset size is small compared to the vision and language domains. If we want to take the protein backbone structure into consideration, that would be reduced to merely 45K pairs of data. Thus, data insufficiency has become the bottleneck of this research problem.”

Both multi-modality and overcoming small datasets are areas of active research. We speculate that some progress can be made by observing ideas used in the machine translation (MT) of rare languages. Meta AI are well positioned to investigate this as leaders in both protein LLM and in MT of rare languages ([Meta, 2023](#)).

### Benchmarks<sup>35</sup>

Ott et al. ([2022](#)) describe the role of benchmarks in progress and innovation: “Benchmarks have become crucial to the development of artificial intelligence (AI). Benchmarks typically contain one or more datasets and metrics for measuring performance. They exemplify and—explicitly or implicitly—define machine learning tasks and goals that models need to achieve. Models achieving new state-of-the-art (SOTA) results on established benchmarks receive widespread recognition. Thus, benchmarks do not only measure, but also steer progress in AI.”

Benchmarks help to identify real progress, Ferruz et al. ([2022](#)) shares an amusing anecdote: “In the first half of the 1990s, at a time when having solved the protein folding challenge sporadically made headlines, the Critical Assessment of Structure Prediction (CASP) set the standard against which in-silico predictive methods for protein structure needed to prove advancement.” Protein folding remained out of the headlines until 2018.

---

<sup>33</sup> Zhang et al. ([2023](#)) we propose to pretrain protein representations according to their 3D structures, yielding significant improvement that synergise well with sequence pretraining.

<sup>34</sup> Notin et al. ([2022](#)) combine autoregressive predictions and homology from retrieved sequences at inference.

<sup>35</sup> For a critique of the state of benchmarks in de novo drug design see Goto ([2021](#)).

## Images and text vs Proteins

We now compare protein generation to image and text generation.

### Objective?

First, the output proteins are objective: they either have the properties we wanted or they do not. This contrasts with judging the performance of systems like DALL-E and ChatGPT, which are so general that they lack (“ground-truth”) objective evaluation.

Objectivity comes at a cost: evaluation requires a wet lab. Costly objective evaluations are thus more suitable for competitions rather than benchmarks. Alongside<sup>36</sup> CASP, CAFA (Critical Assessment of Function Annotation), and CAPRI (Critical Assessment of PRediction of Interactions) are bi-annual contests in which many research groups predict structures, functions or interactions of proteins not yet (publicly) experimentally determined.

Benchmarks, unlike competitions, are accessible at any given point in time and can be iterated upon. This makes them crucial for continuous research and progress.

### Unintuitive?

DALL-E and ChatGPT still allow for a somewhat accurate<sup>37</sup> *subjective impression* of impressiveness. But with protein design we are in the dark: objectivity is the model, while I alone cannot tell whether a generated protein is toxic or virulent. So detecting progress is naturally more difficult.

### Proxies?

Imperfect evaluations have proved useful in visual and textual domains: measurements such as the Frechet Inception Distance ([Huesel et al., 2017](#))<sup>38</sup> proved to be useful in the development of image models.

### Downstream tasks?

In 2019, in the infancy of protein representation learning, there were no standardized benchmarks. Rives et al. ([2021](#)) and Bepler et al. ([2019](#)) report incomparable transfer learning results. TAPE (Tasks Assessing Protein Embeddings) introduced by Rao et al. ([2019](#)) was the first set of diverse tasks in a convenient, standardized format which was aimed to assist protein engineering.

---

<sup>36</sup> Note that all of these focused on prediction and not generation. We are not aware of “objective” protein generation competitions.

<sup>37</sup> Bowman ([2023](#)) cautiously notes: “Brief interactions with LLMs are often misleading” as clever prompt engineering often uncovered capabilities unavailable at the first sight.

<sup>38</sup> FID rates model highly even if it just memorized the data, failing to produce novel images.

Later other datasets were developed:

- Huang et al. (2021) introduced TDA, an extensive evaluation platform, containing protein-related datasets and tasks for drug discovery
- Townshend introduced (2021) ATOM3D provides benchmark datasets for 3D molecular learning tasks.
- Dallago C, et al. (2022) introduced FLIP, a three protein landscape benchmarks for fitness prediction evaluation.
- Capel et al. (2022) introduced ProteinGLUE, “a set of seven tasks for evaluating learned protein representations.”
- Xu et al. (2022) introduced PEER, “a set of diverse protein understanding tasks including protein function prediction, protein localization prediction, protein structure prediction, protein-protein interaction prediction, and protein-ligand interaction prediction.”
- Notion et al. (2022) introduced [ProteinGym](#), “an extensive set of multiplexed assays of variant effects, substantially increasing both the number and diversity of assays compared to existing benchmarks.”

In private communications researchers report issues with data contamination, particularly issues with properly separating training data from test data due to many biological sequences being very similar to each other as they are related by evolution ([Petti & Eddy, 2022](#)).

Liu et al. (2023) notes that “for some types of predictive tasks, like stability, independent simulation-based methods, such as Rosetta, could complement the surrogate-based evaluation.”

Zheng et al. (2023) argues that in other domains, the successful recovery of native data (and their in-domain held-out set), coupled with the ability to synthesize completely new data, was enough for generative models to capture underlying patterns and generalize.

## But do they actually generate proteins?

Nonetheless, with so many synthetic benchmarks and evaluations focused on reconstruction, we are left wondering if protein generation methods really work.

They do.<sup>39</sup> Madani et al. (2021) experimentally evaluate model-generated artificial proteins confirming ability to perform de novo protein generation. Verkuil et al. (2022) “focus on two protein design tasks: fixed backbone design where the structure is specified, and unconstrained generation where the structure is sampled from the model” and find “high overall success rates (152/228 or 67%) in producing a soluble and monomeric species by size exclusion chromatography” while observing that “35 [of 152 design] have no significant sequence match to known natural proteins” showing that it’s not about memorisation. And “Lorenz [CTO of [Basecamp Research](#)] says that his team’s own [protein] design experiments have achieved an 80% success rate at producing functional proteins” ([Eisenstein, 2023](#)).

Zheng et al. (2023) summarized the state of indirect evidence well: “these protein language models are able to generalize across a wide range of downstream applications and can

<sup>39</sup> Outside of protein design, we are aware of the following work where AI-generated molecules have been experimentally validated: [Polykovskiy et al., 2018](#), [Merk et al., 2018](#), [Merk et al., 2018](#), [Zhavoronkov et al., 2019](#), [Tan et al., 2020](#), [Li et al., 2020](#), [Yang et al., 2020](#), [Yoshimori et al., 2020](#), [Perron et al., 2022](#), [Korshunova et al., 2022](#), [Godinez et al., 2022](#).

capture evolutionary information about secondary and tertiary structures from sequences alone. They have recently been demonstrated with strong capabilities in uncovering protein structures, predicting the effect of sequence variation on function, antibody infilling, and many other general purposes.”

## Putting it all together for biology

For a final forecast, we decompose the trajectory of AI solving a domain into two steps:

1. an “ImageNet moment” (“a model, dataset and pretraining task that provide strong off-the-shelf performance for most tasks, even with little data” to quote [Ofer et al. 2021](#)), followed by
2. impressive<sup>40</sup> generative models first being trained.

Finally, we guess how fast these models will proliferate under a business-as-usual scenario.

### ImageNet moment

Current models are capable of impressive deeds like protein folding prediction, massively expanding the speed at which data on a sequence is made available for further work.<sup>41</sup>

Nonetheless, the models are not multimodal enough, and nor have they demonstrated learning from a diverse set of small datasets. At the same time, due to fast scaling, we should expect more from the next generation of pre-trained protein language models than we otherwise would.

It seems highly likely that AIs will assist humans with AI research a few years from now, given interest from both Meta, who have relevant expertise, and Deepmind, who have demonstrated breakthrough after breakthrough. And so it’s hard to see why a suitably scalable multimodal architecture would take more than 1–5 years to develop.

Another possibility is that the most impressive LLMs will not only be trained on visual and textual internet data but also on other modalities, like protein sequences. This huge increase in the available training data could enable a discontinuous jump in model size, and hence (assuming scaling continues to work) in capabilities.

### Impressive generation

As a reminder, image generation progressed from “underwhelming” to “very impressive” in 7 years; NLP went from its “ImageNet moment” to ChatGPT in 4–5 years. We are already past the “underwhelming” generation of protein-LMs, as we can synthesize actual proteins (see [But do they actually generate proteins?](#)).

---

<sup>40</sup> Here “impressive” is of course imprecise, we are roughly tracking “widely used by the general public because of how good it is.”

<sup>41</sup> ESM-2 powers ESM-Fold ([Lin et al., 2022](#))

In [Algorithms](#), we discussed ways in which protein generation is clearly behind on adopting NLP's best practices. Biologically-aware directions (e.g. domain-specific tokenizers) could result in significant improvements (and Ankh ([Elnaggar et al., 2023](#)) has already delivered some of them, leading to 10x savings in model size). On willingness to scale and develop technology: we see great interest from leading AI labs to continue contributing to science – and on the other hand, we have pharma companies, who have a strong financial incentive in the success of protein generation. We should thus expect a lot of algorithmic progress.

On the other hand, good quality data seems to be limited to UniRef50. This unavailability of data is likely to affect how much scaling can be done. We should expect more data to become available as this is commonly realized. We see several ways the bottleneck can be revealed:

1. A growing bioinformatics market and growing investment in AI for drug discovery ([Future willingness to spend](#)) will likely create a niche for companies specializing in creation of proprietary datasets specifically for generative AI. And better quality data can significantly improve performance<sup>42</sup>. Some of these companies would likely specialize in synthetic data to enlarge particularly small datasets.
2. Expand protein sequence data might be to 2.3B genomic sequences from WGS.
3. Lastly, it doesn't seem impossible that progress in AI could result in improved modeling/simulation leading to much more "pseudo-experimental" data that previously was bottlenecked by wet labs<sup>43</sup>.

Another factor that could alleviate the data bottleneck is the growing cost of leading NLP systems. Sam Altman claimed that GPT-4 cost >\$100M, with future systems likely to cost even more. At this point, it's already economical for leading labs like OpenAI to put a lot of research talent into increasing sample efficiency, which quickly trickles down to protein ML.

A smaller factor is the rather uninspiring state of relevant benchmarks ([Benchmarks](#)); see also Goto ([2021](#)). While generative AI in general can progress without good benchmarks (e.g. computer's use of indirect proxy metrics), benchmarks are extremely helpful. Given the current state of AI technology, we wouldn't be surprised to see the automated creation of benchmarks, resolving this issue.

Overall, data and benchmarks are presently notably worse than the counterparts in early CV and NLP. Protein ML will likely compensate for that with willingness to spend and algorithmic progress (partly through transferring ideas from the sequence modelling leader, NLP).

Following protein's Imagenet moment, we predict it will take 2–4 years to get towards a really impressive protein generation, with a longer tail if its data bottleneck is not relieved.

---

<sup>42</sup> See sections [Issues with data quality](#) for a protein-specific example. And [Could the problem go away?](#)

<sup>43</sup> We already saw that to some degree in NLP, see Jiang ([2023](#)).



## Proliferation through open-source

We discussed the hyperactivity of the open-source ML community and how little they lag the leading labs (in part due to some of the leading labs regularly open-sourcing their models).

The situation is seemingly worse in protein generation, *all* three most capable models were open-sourced on day one:

- Ankh ([Elnaggar et al., 2023](#)) @ <https://github.com/agemagician/Ankh>
- ESM-2 ([Lin et al., 2022](#)) @ [github.com/facebookresearch/esm](https://github.com/facebookresearch/esm)
- ProGen2 ([Nijkamp, 2022](#)) @ [github.com/salesforce/progen/tree/main/progen2](https://github.com/salesforce/progen/tree/main/progen2) Meta AI

and Salesforce Research seem to be committed to open access:

- In their review of evolution of LLMs, Yang et al. ([2023](#)) write “Meta contributes significantly to open-source LLMs and promotes research of LLMs. When considering contributions to the open-source community, particularly those related to LLMs, Meta stands out as one of the most generous commercial companies, as all the LLMs developed by Meta are open-sourced.”
- Salesforce Research also open sourced their other major models, such as CodeGen ([Nijkamp et al., 2022](#)) and CodeGen2 ([Nijkamp et al., 2023](#)).

On the other hand, while the open-source community got excited about StableDiffusion and ChatGPT, they seem less inclined to get excited about powerful biological models. We put this down to the unintuitive domain: human hobbyists can’t distinguish protein sequences from each other. Most people most likely wouldn’t be able to synthesize anything given a sequence (in part due to KYC policies). One cannot say that interest is negligible though: ESM-2 has 2k Github stars. T5 (for years the most powerful open-sourced LLM) has 5.2k stars.

Either way, with the labs keeping their default open-source-everything policy, it’s hard not to see insights proliferating. It’s unclear how the labs would realize it is a good time to stop this practice<sup>44</sup> – especially if we take into account AI progress was surprising and it has been quite possible simply to not notice latent AI capabilities (see Bowman ([2023](#))). If a generation of models just short of “extremely impressive” is open-sourced, it seems highly likely that someone in open source would pull it over the threshold.

Proliferation is likely to be instantaneous unless norms change (they might – there is growing concern about dual-use ([Urbina et al., 2022](#))), or if accessing key datasets would come with a legal requirement to release the resulting model checkpoints. In this case, proliferation might be delayed by some years.

---

<sup>44</sup> One option is that they would licence proprietary datasets with licences limiting how model checkpoints could be shared.

## In sum

- Protein's ImageNet moment might arrive in 1–5 years given attention from Meta and DeepMind, and with AIs accelerating AI research.
- Following protein's Imagenet moment, it will take 2–4 years to get towards a really impressive protein generation with a longer tail if data bottleneck wouldn't be relieved.
- Proliferation is likely to be instantaneous unless norms to open access will change or if access to key datasets will require not to release the model checkpoints. Otherwise, it might be delayed by some years.

This means that impressive generative capabilities are likely to be developed in 3 to 8 years.