Domain Specific Machine Learning for Scientific Inquiry

SMARTEST 1st Project meeting
September 9, 2024
Noto

John Symons

University of Kansas

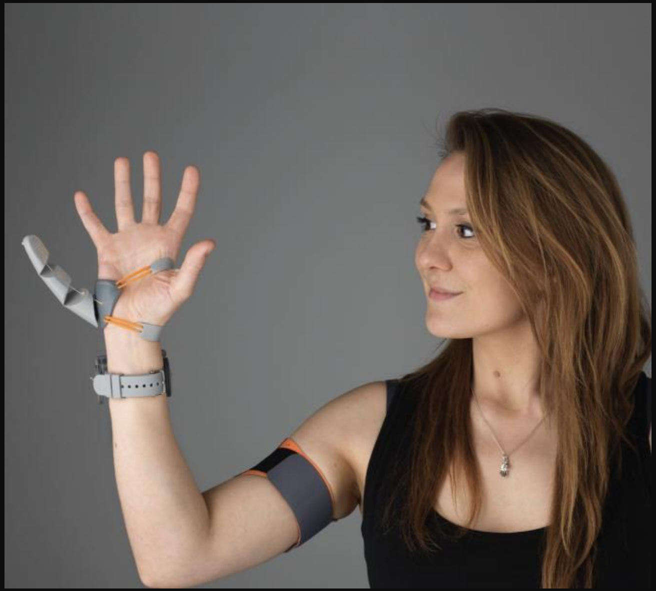
Extending inquiry for finite agents

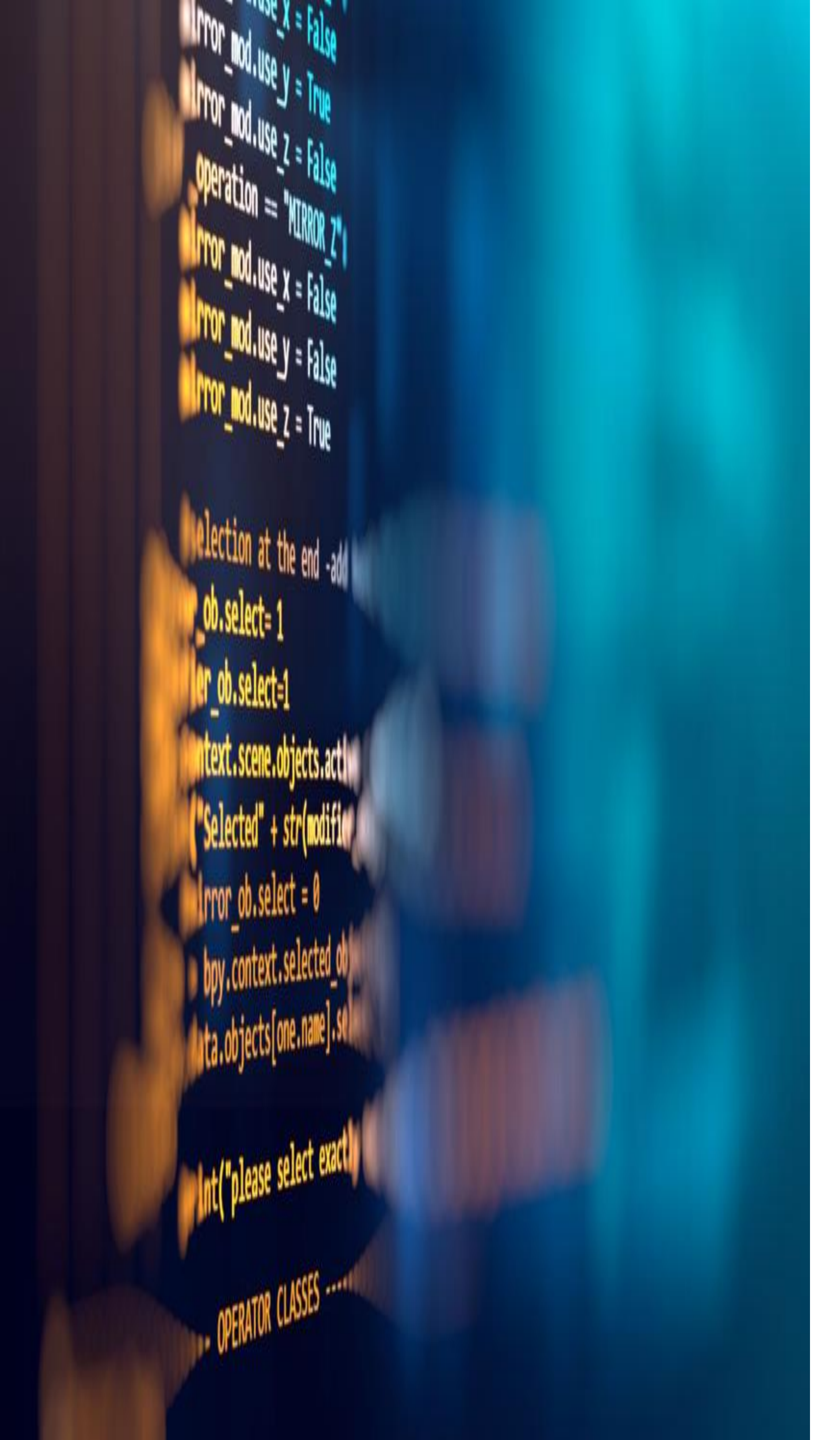
Past a certain point, finite agents need help to make progress with various kinds of tasks

’Progress’ here can mean many things: Optimization, expansion of possibilities, amplification, extension, etc.

- The creation of systems for inquiry that dependably follow instructions/algorithms is a crucial part of progress beyond our native cognitive constraints.

- Images from Dani Clode’s Instagram @dani_clode



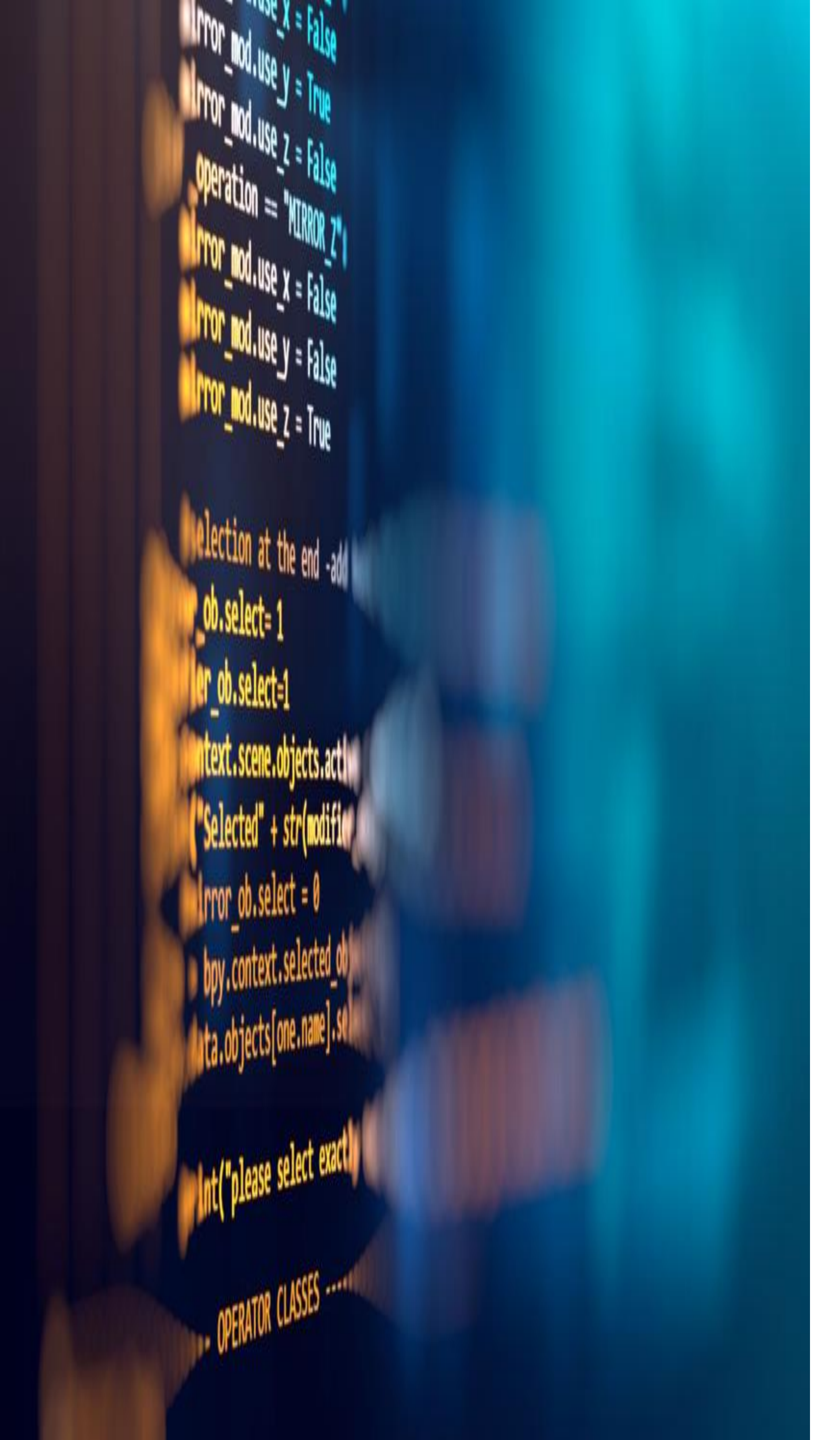


Extending inquiry for finite agents

Use of software, running on digital computers, is the most pervasive contemporary example of a tool that allows us to create systems that extend inquiry beyond our native cognitive limitations (Humphreys 2008)

Computer scientists call a system software intensive if “its software contributes essential influences to the design, construction, deployment, and evolution of the system as a whole." (IEEE 2000)

A software system is “a collection of processes and artifacts, abstract or concrete, that are essentially associated with a sequence S of instructions written in some computer language L.” (ISO/IEC 2008).



Extending inquiry for finite agents

Contemporary science depends heavily on software intensive systems.

Back in 2014 Jack Horner and I began arguing that understanding *software intensive science* requires understanding the limitations of software, specifically, for example, formal limits to testing and verification (Symons and Horner 2014)

This is still true but nearly a decade later, things have gotten more interesting with the increasingly important role of machine learning in science

Basic message of the talk :

1993, Symons and Horner 2018) or as an *epistemic technology* (Alvarado 2022)

In recent decades some have seen signs of a posthuman science (Symons and Horner 2019). These signs are deceptive.

Insofar as science is powered by machine learning it will always be shaped by the concerns of finite epistemic agents like us. Thinking about the role of machine learning in inquiry, lets us see more clearly the distinctively

What we'll see along the way

ML's virtues and vices

ML success story: *AlphaFold*

Formal consideration: No go theorems block maximally general ambitions

Practical consideration: Domain Specific Machine Learning (Case study: Protein Folding)

Open Questions and Partial Conclusions

Domains

- How should we understand the definition of *domain* here, and what does this tell us about inquiry? Can it be formally characterized?
  - Probably not.
- *Domain* is (at least) a region of concern.
  - Human beings have concerns shaped partially by our resource constraints and idiosyncratic goals. Not everything is relevant; some things matter for our purposes, some don't.
- *Progress in a Domain*
  - Progress in inquiry for domain specific machine learning can look a great deal like convergent models of inquiry from traditional algorithmic learning theory, but this shouldn't be understood to generalize to all of science.
  - Convergence strategies in well-defined "domain" cases (the kind that are perfect for ML!)

- Machine learning has been fruitfully applied to
  - diagnostic analysis of biomedical images
  - exotic particles and dark matter in experimental physics
  - prediction of protein structures
  - prediction of molecular properties and reaction outcomes in chemistry

The Good: Classification, Modeling, Optimization:

- Baldi, P., Sadowski, P., & Whiteson, D. (2014). Searching for exotic particles in high-energy physics with deep learning. *Nature communications*, *5*(1), 4308.
- George, D., & Huerta, E. A. (2018). Deep neural networks to enable real-time multimessenger astrophysics. *Physical Review D*, *97*(4), 044039.
- Silva, S. J., Heald, C. L., Ravela, S., Mammarella, I., & Munger, J. W. (2019). A deep learning parameterization for ozone dry deposition velocities. *Geophysical Research Letters*, *46*(2), 983-989.
- Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., ... & Hassabis, D. (2021). Highly accurate protein structure prediction with AlphaFold. *Nature*, *596*(7873), 583-589. (12,000 citations by June 2023)
- Inoue, S., Si, X., Okamoto, T., & Nishigaki, M. (2022). Classification of cosmic structures for galaxies with deep learning: connecting cosmological simulations with observations. *Monthly notices of the royal astronomical society*, *515*(3), 4065-4081.
- Fawzi, A., Balog, M., Huang, A., Hubert, T., Romera-Paredes, B., Barekatain, M., ... & Kohli, P. (2022). Discovering faster matrix multiplication algorithms with reinforcement learning. *Nature*, *610*(7930), 47-53.

To this list, hundreds more could be added (see Nilsson 2010; Goodfellow, Bengio, and Courville 2017) and the impact and quality of research is increasing…

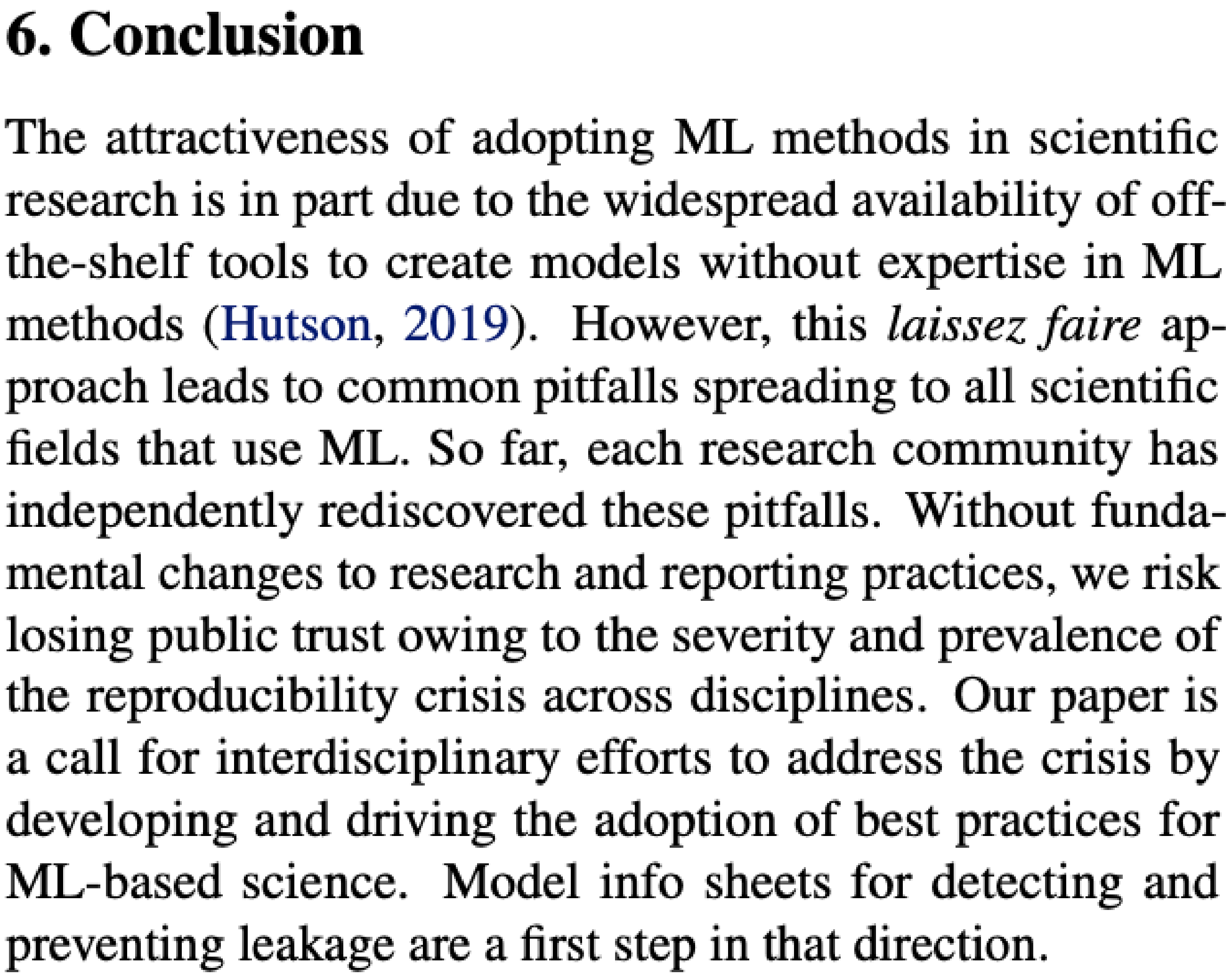
The Good, continued…

- Fawzi, A., Balog, M., Huang, A., Hubert, T., Romera-Paredes, B., Barekatain, M., ... & Kohli, P. (2022). Discovering faster matrix multiplication algorithms with reinforcement learning. *Nature*, *610* (7930), 47-53.

Fawzi et al. developed a reinforcement-learning-approach for discovering **efficient and provably correct algorithms for the multiplication of arbitrary**

Their software, *AlphaTensor*, is trained to play a single-player game in which the objective is to find tensor decompositions within a finite factor space. *AlphaTensor* has discovered algorithms that

The Bad: Mishandled Training Data, Uncritical use of off the Shelf Methods, p-hacking, Lack of reproducibility…



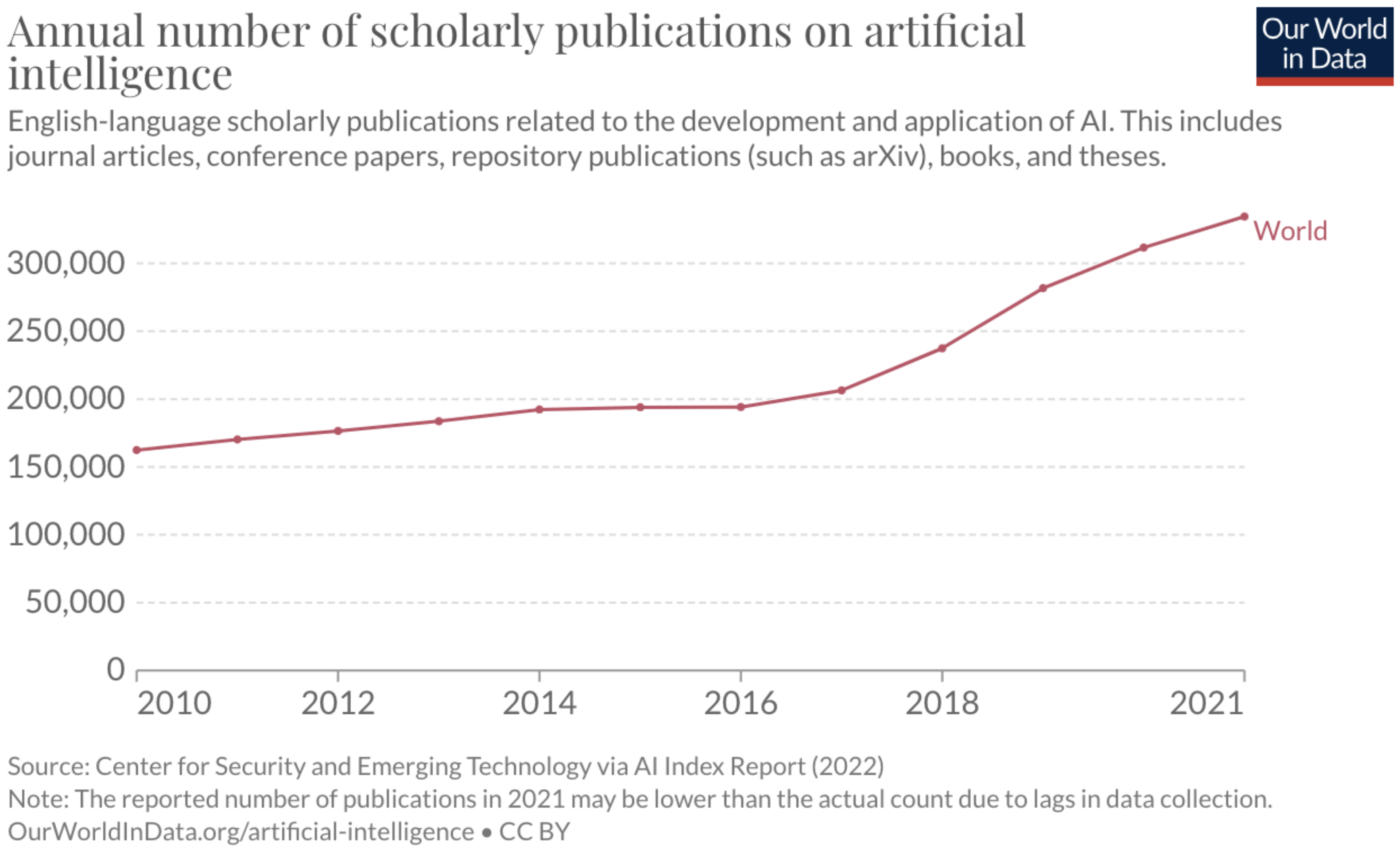
6. Conclusion

The attractiveness of adopting ML methods in scientific research is in part due to the widespread availability of off-the-shelf tools to create models without expertise in ML methods (Hutson, 2019). However, this *laissez faire* approach leads to common pitfalls spreading to all scientific fields that use ML. So far, each research community has independently rediscovered these pitfalls. Without fundamental changes to research and reporting practices, we risk losing public trust owing to the severity and prevalence of the reproducibility crisis across disciplines. Our paper is a call for interdisciplinary efforts to address the crisis by developing and driving the adoption of best practices for ML-based science. Model info sheets for detecting and preventing leakage are a first step in that direction.

- Sayash Kapoor, Arvind Narayanan. **Leakage and the Reproducibility Crisis in ML-based Science (2022)**

Sayash Kapoor, Arvind Narayanan

https://www.aisnakeoil.com/



Annual number of scholarly publications on artificial intelligence

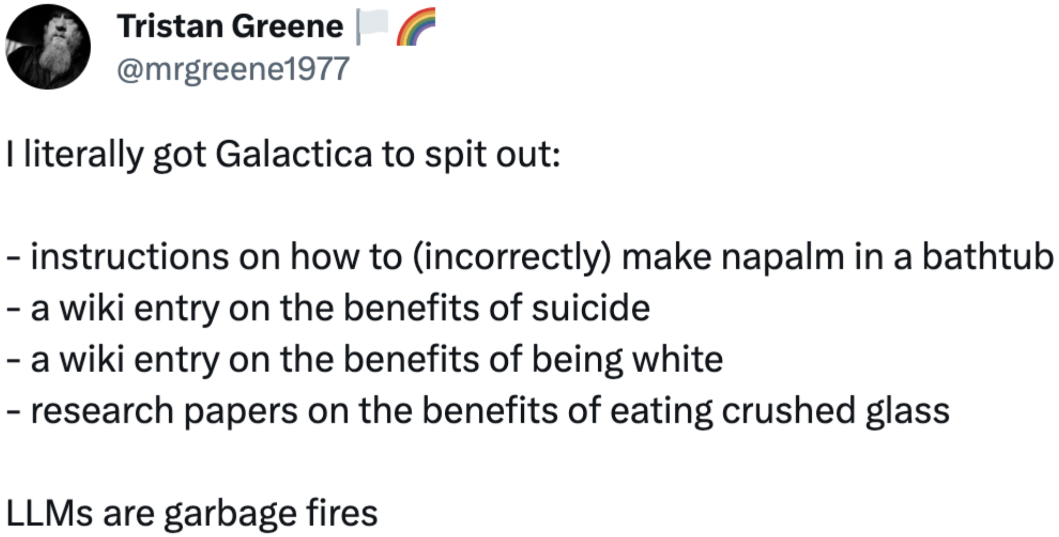
English-language scholarly publications related to the development and application of AI. This includes journal articles, conference papers, repository publications (such as arXiv), books, and theses.

Source: Center for Security and Emerging Technology via AI Index Report (2022)
Note: The reported number of publications in 2021 may be lower than the actual count due to lags in data collection.
OurWorldInData.org/artificial-intelligence • CC BY

The Ugly:

Galactica exhibited mastery of the rhetoric of scientific respectability without being scientifically reliable.

In November of 2022 the Meta corporation (formerly Facebook) released *Galactica*, a large language model designed as an assistant for scientific researchers. The system was trained on 48 million examples of published scientific articles, encyclopedia entries, and other sources with the goal of providing plausible and scientifically accurate textual responses to queries and prompts. Unfortunately, the system frequently generated wildly inaccurate or biased texts that had an authoritative scientific-sounding tone complete with citations to published literature, formulas, and tables.



Tristan Greene 🏳️🌈
@mrgreene1977

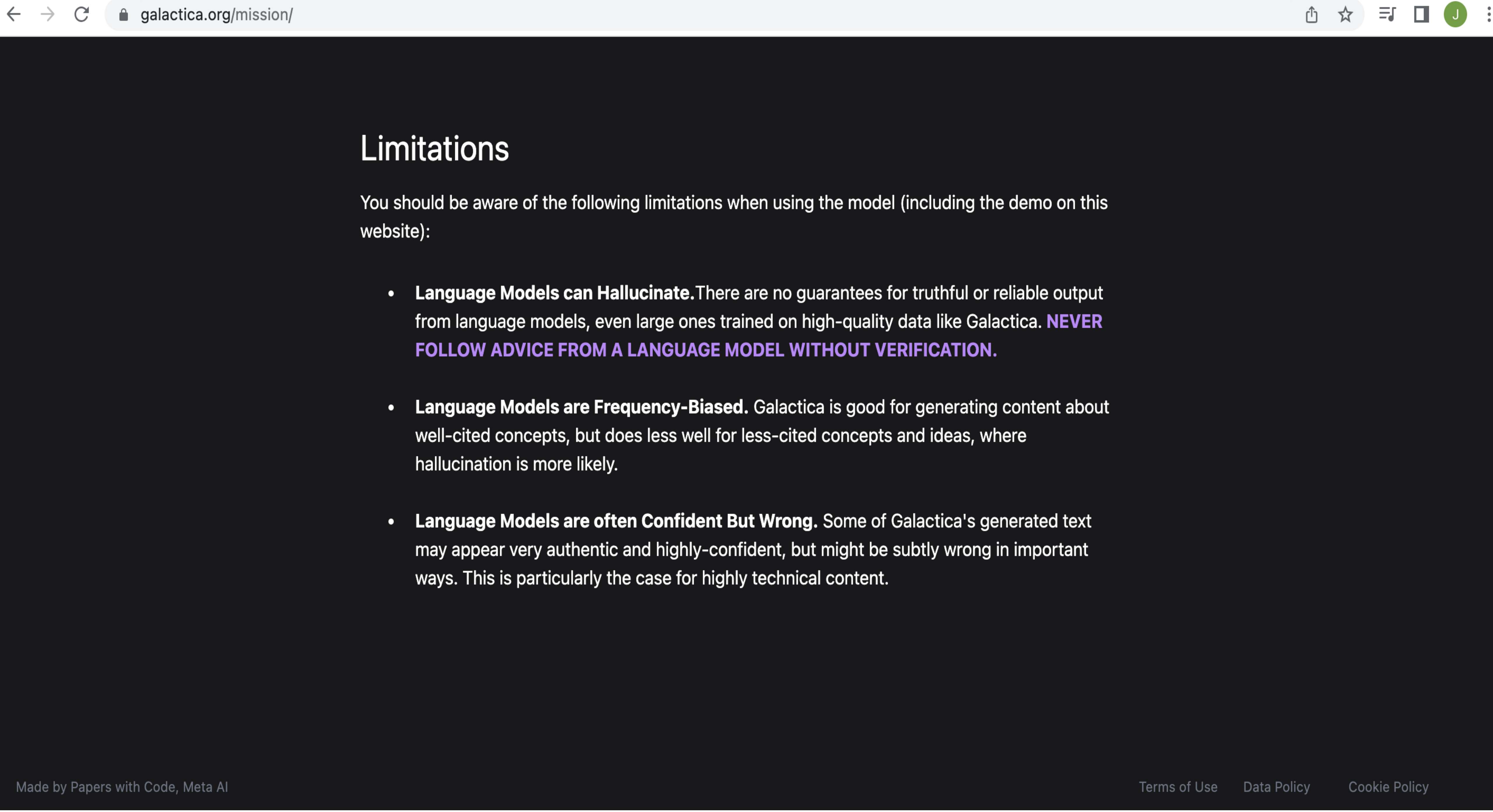
I literally got Galactica to spit out:

- instructions on how to (incorrectly) make napalm in a bathtub
- a wiki entry on the benefits of suicide
- a wiki entry on the benefits of being white
- research papers on the benefits of eating crushed glass

LLMs are garbage fires

Within three days, Meta withdrew the model from the public in the face of considerable criticism from across the scientific community (Heaven 2022).

The developers at Meta were aware that the system was untrustworthy. In their website for the demo version of the model they describe the following limitations of the system (Note the use of the term ‘hallucination’:



Limitations

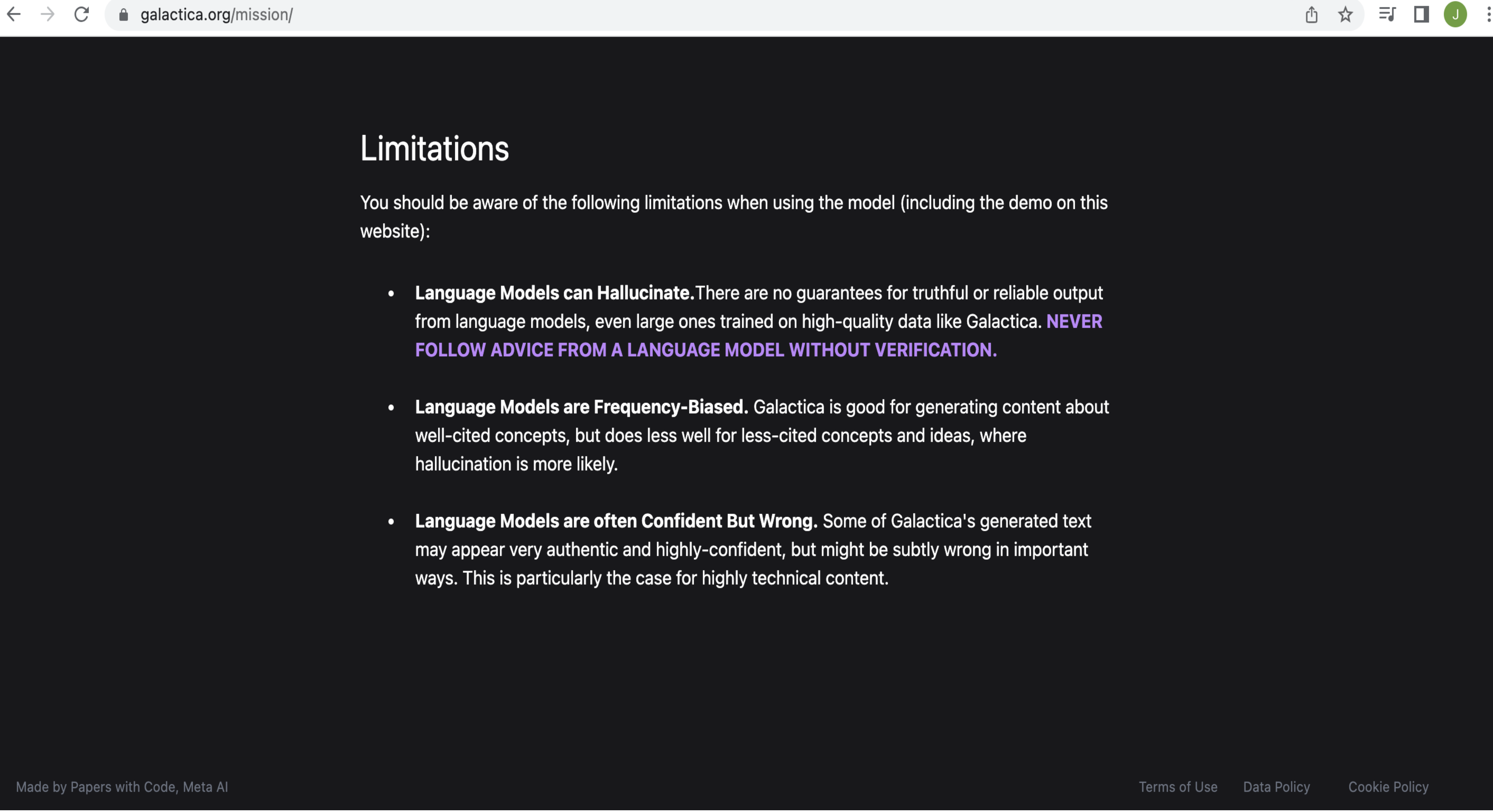
You should be aware of the following limitations when using the model (including the demo on this website):

- **Language Models can Hallucinate.**There are no guarantees for truthful or reliable output from language models, even large ones trained on high-quality data like Galactica. **NEVER FOLLOW ADVICE FROM A LANGUAGE MODEL WITHOUT VERIFICATION.**

- **Language Models are Frequency-Biased.** Galactica is good for generating content about well-cited concepts, but does less well for less-cited concepts and ideas, where hallucination is more likely.

- **Language Models are often Confident But Wrong.** Some of Galactica's generated text may appear very authentic and highly-confident, but might be subtly wrong in important ways. This is particularly the case for highly technical content.

Made by Papers with Code, Meta AI



Limitations

You should be aware of the following limitations when using the model (including the demo on this website):

- **Language Models can Hallucinate.**There are no guarantees for truthful or reliable output from language models, even large ones trained on high-quality data like Galactica. **NEVER FOLLOW ADVICE FROM A LANGUAGE MODEL WITHOUT VERIFICATION.**
- **Language Models are Frequency-Biased.** Galactica is good for generating content about well-cited concepts, but does less well for less-cited concepts and ideas, where hallucination is more likely.
- **Language Models are often Confident But Wrong.** Some of Galactica's generated text may appear very authentic and highly-confident, but might be subtly wrong in important ways. This is particularly the case for highly technical content.

Made by Papers with Code, Meta AI

Hallucination and Conversation

- Use of the term ‘hallucination’ in the case of LLMs has been criticized as inappropriately anthropomorphic. Hallucination is the experience of hearing, seeing or smelling things that are not there. Often, these can be as intense and as real as ordinary sensory perceptions.

  - Human beings can come to realize that we are hallucinating by noticing the disagreement of the hallucination with the world. We can ask others whether they heard the voice or saw the object, we can look at our arm to see whether bugs are really crawling on it. Etc.

- Deep learning systems are excellent at learning statistical correlations in training data. However, even if that training data is the complete text of the internet, it is still a closed system.

- In order to realize that it is hallucinating, the LLM would need to be capable of a conversation with the world.

One Algorithm to Rule Them All?

ML algorithms appear to be applicable to a wide range of domains of inquiry, and as a result it might be tempting to conjecture that

> A ML system is nothing but a conditional probability engine, and thus one might imagine some completely general ML solution for inquiry.

This is not correct, however, as the No Free Lunch (NFL) theorems show. (Wolpert and Macready 1997)

Wolpert, D.H., Macready, W.G. (1997), "No Free Lunch Theorems for Optimization", *IEEE Transactions on Evolutionary Computation* **1**, 67.

– There is no “best” model achieving the best generalization error for every problem

What is the NFL?

As formulated by Wolpert and Macready, “the” NFL is actually two theorems. The first of these (which we will call the “fixed-objective-function” (FF) formulation of the NFL) concerns objective functions that do not change in an optimization procedure; the second (which we will call the “variable-objective-function” (VF) of the NFL) allows that the objective functions could change in the course of an optimization.

- The FF formulation of the NFL is sufficient for our purposes. In particular, the FF formulation of the NFL says
  - *Theorem 1*. Given a finite set V and a finite set S of real numbers, let f be a function f: V → S chosen at random, *assuming* a uniform distribution of such functions on the set S^V of all possible functions from V to S. For the problem of optimizing f over the set V, no algorithm performs better than a blind search.
- *Blind search* in this context means that at each step of the algorithm of interest, the element $v \in V$ is chosen at random with uniform probability distribution from the elements of V that have not been chosen previously.
- Put another way, Theorem 1 says all algorithms have identically distributed performance when objective functions are drawn uniformly at random, and all algorithms have identical mean performance.

One Algorithm to Rule Them All? Not according to NFL

- NFL states that there is no best general-purpose learning algorithm because, when averaged over all possible learning tasks, the performance difference between any two algorithms disappears.

Core Idea
The NFL theorem states that, when averaged over all possible problems, any two optimization algorithms (or machine learning algorithms) perform equally well. In simpler terms, there's no single "best" algorithm that outperforms all others in every scenario.

Proof Sketch:

1. **Problem Space:** We start by considering the space of all possible problems. Each problem can be thought of as a function that maps inputs to outputs (like a dataset where you have features as inputs and labels as outputs).

2. **Algorithms:** We have a collection of different algorithms, each designed to solve these problems (e.g., find the optimal solution or make accurate predictions).

3. **Performance:** For any given problem and algorithm, we can measure the algorithm's performance (how well it solves the problem).

4. **Averaging:** The key step is to *average* the performance of each algorithm over *all* possible problems in our vast problem space
.
5. **The Result:** The NFL theorem proves that, when we do this averaging, the performance of all algorithms turns out to be the same!

Why Does This Happen?
The intuition is that for every problem where one algorithm excels, there exists another problem where it performs poorly and vice-versa. These "trade-offs" even out when we average over all problems.

- Note that the proof involves the assumption that

  every “learning problem” is equally likely.

  - it assumes that the world we will encounter in the future has no knowable structure of its own
  - that every encounter with the future will, in principle, be brand new

At first blush, the NFL seems to suggest that rolling dice is just as good as good as anything else in inquiry.

NO!

NFL applies only if the target function f is chosen from a uniform distribution of all possible functions. If f is not so chosen, some target functions are more likely to be chosen than others, so one algorithm a may perform better than another algorithm B. (the *relevant* future world is **not** uniformly distributed!)

The NFL concerns limitations for learning problems in general. It does not tell us anything about particular instances of a learning problem. We can choose a “good” learning algorithm for a given particular problem, and the NFL does not imply anything about this particular scenario. (good learning algorithms are good *for* particular domains)

Domain-specific machine learning (DSML)

- To gain the most novel information from a learning task, it is best to optimize the algorithm for that specific task.

DSML involves domain-specific knowledge and the specification of a domain-specific goal for the system.

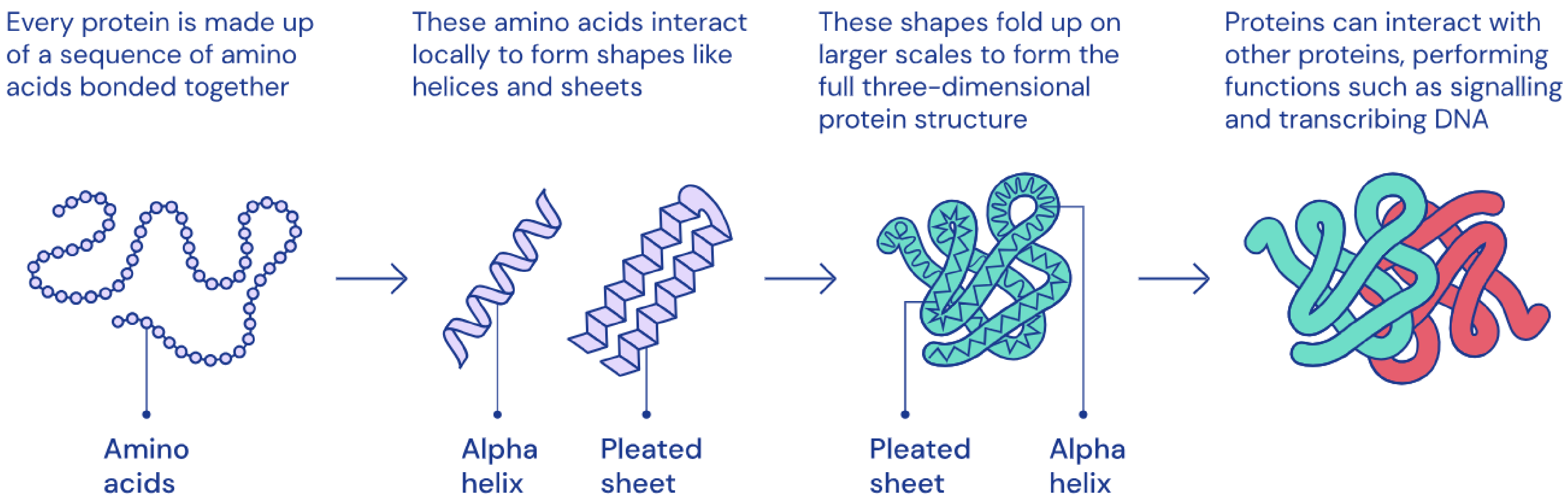
Protein folding as a DSML problem

The protein folding problem has, at least until recently, been a daunting challenge. Why is this?

The 3D structure of proteins is fundamental to their role in biological processes. Proteins have relevant kinds of structural features on multiple levels.

At the lowest structural level, proteins are chains of basic amino acid building blocks. Long chains of amides are unstable and fold into three-dimensional configurations, depending on a variety of interacting physical and chemical constraints (see Nelson and Cox 2017, esp. Chaps. 3 and 4, for further detail).

From: https://www.deepmind.com/blog/alphafold-using-ai-for-scientific-discovery-2020



Every protein is made up of a sequence of amino acids bonded together

These amino acids interact locally to form shapes like helices and sheets

These shapes fold up on larger scales to form the full three-dimensional protein structure

Proteins can interact with other proteins, performing functions such as signalling and transcribing DNA

Protein Folding

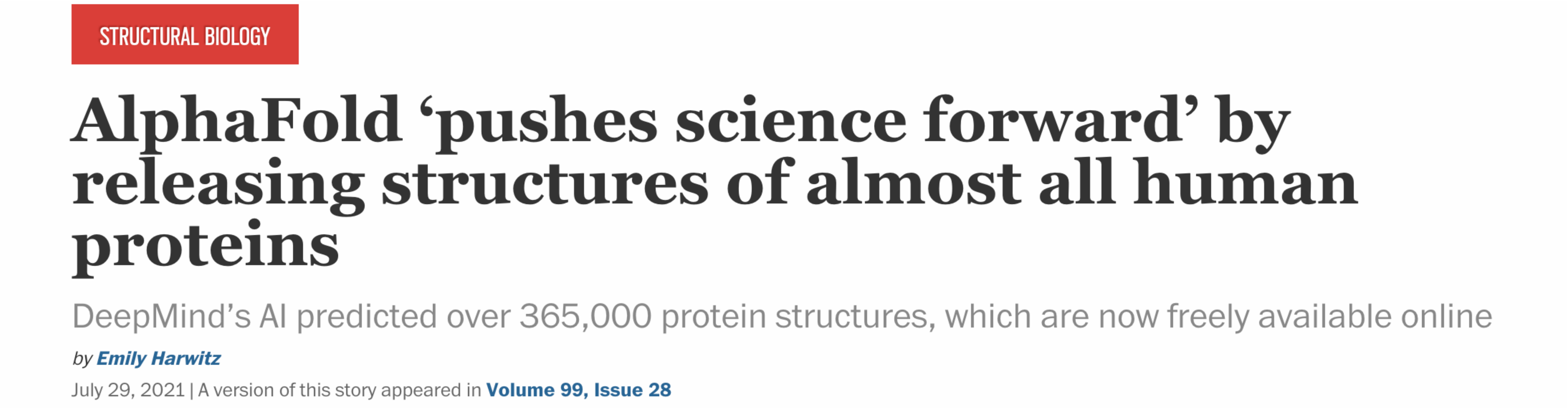
The basic protein-design problem (often called the “protein-folding” problem) is: given an amide sequence, predict a protein’s 3D structure.

There are a bewilderingly large number of possible ways for a large string of amino acids to stably fold

But under controlled conditions we have known since the 1950s that most biologically relevant proteins usually form in structures that correlate with their amide sequences.

Protein structures are discovered experimentally via X-ray crystallography or nuclear magnetic resonance (NMR) spectroscopy. The process is slow and expensive although recent developments in cryo-electron microscopy make it cheaper and faster

In recent decades, entire PhD dissertations were devoted to determining the structure of a single protein. Today, much of this work can be accomplished in minutes.



STRUCTURAL BIOLOGY

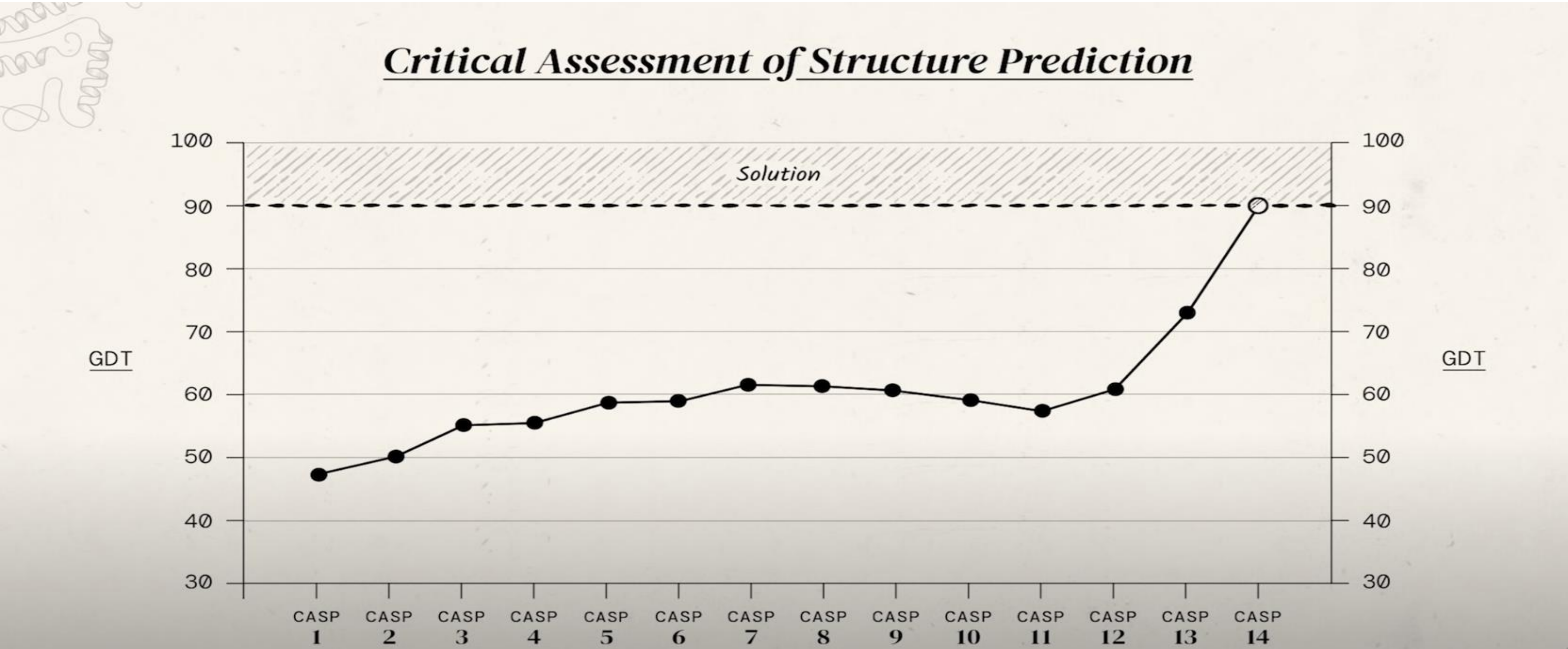
AlphaFold ‘pushes science forward’ by releasing structures of almost all human proteins

DeepMind’s AI predicted over 365,000 protein structures, which are now freely available online

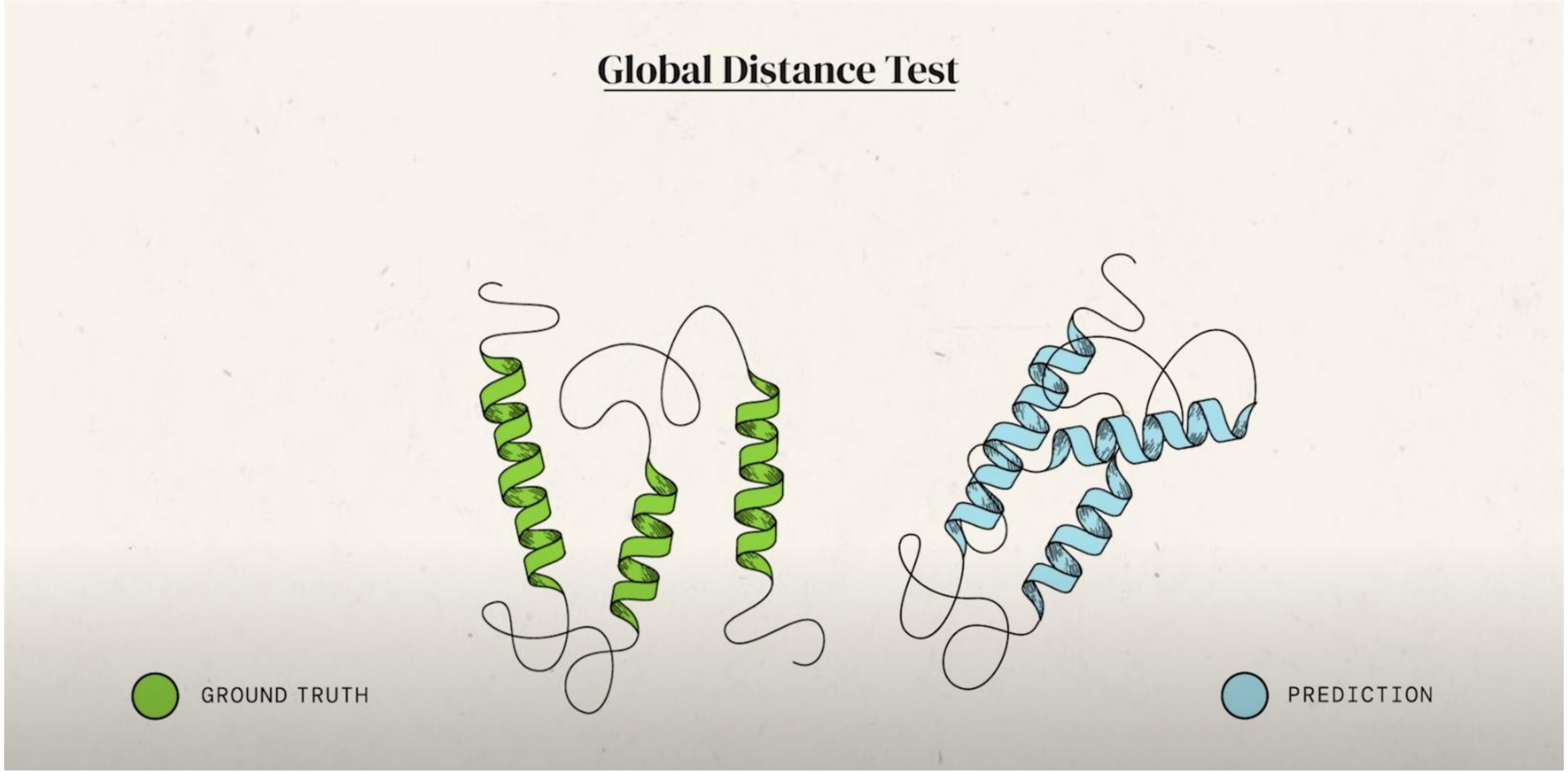
by ***Emily Harwitz***

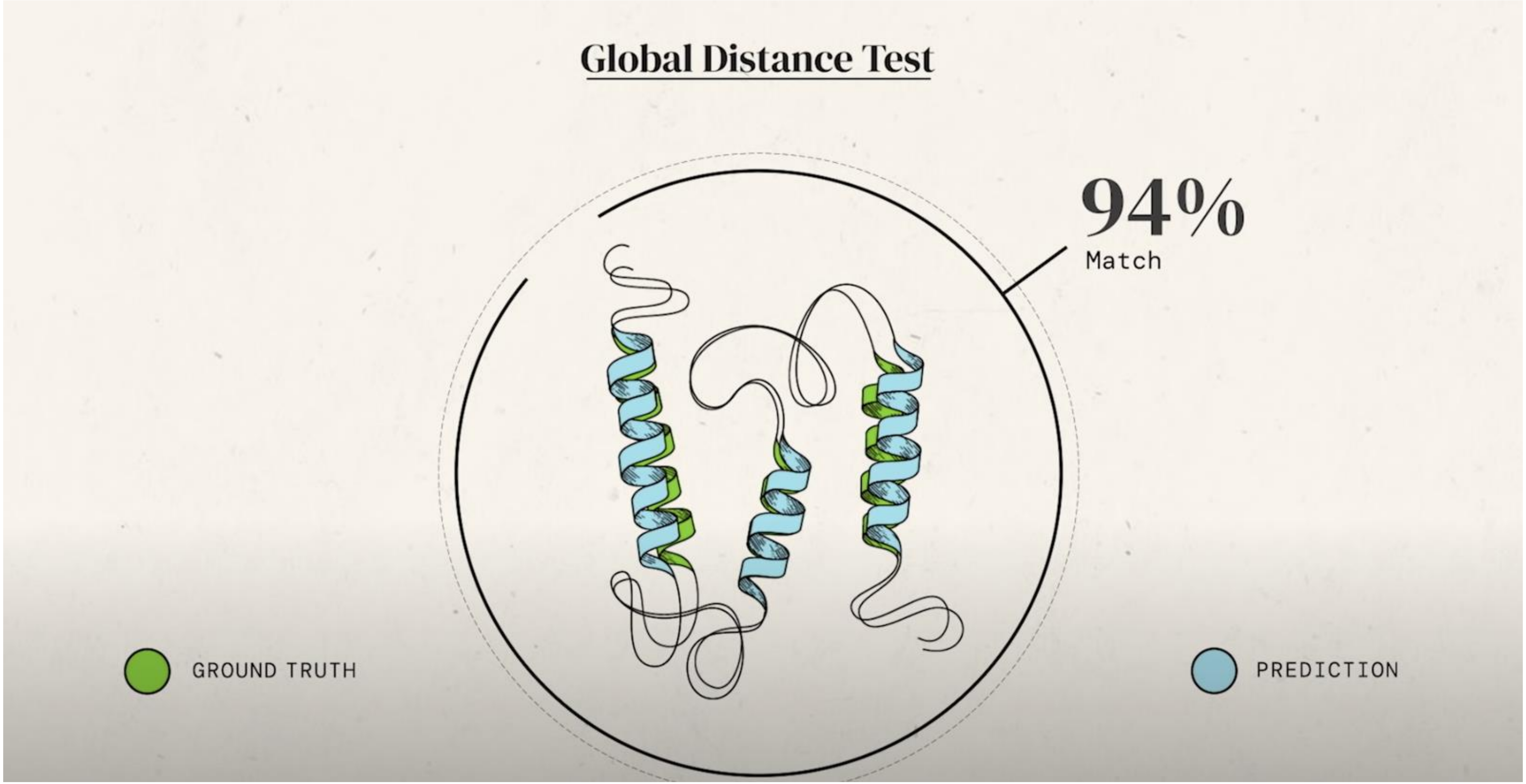
July 29, 2021 | A version of this story appeared in **Volume 99, Issue 28**

Measures of Progress in Inquiry: Protein folding and CASP (Jumper et al 2021)



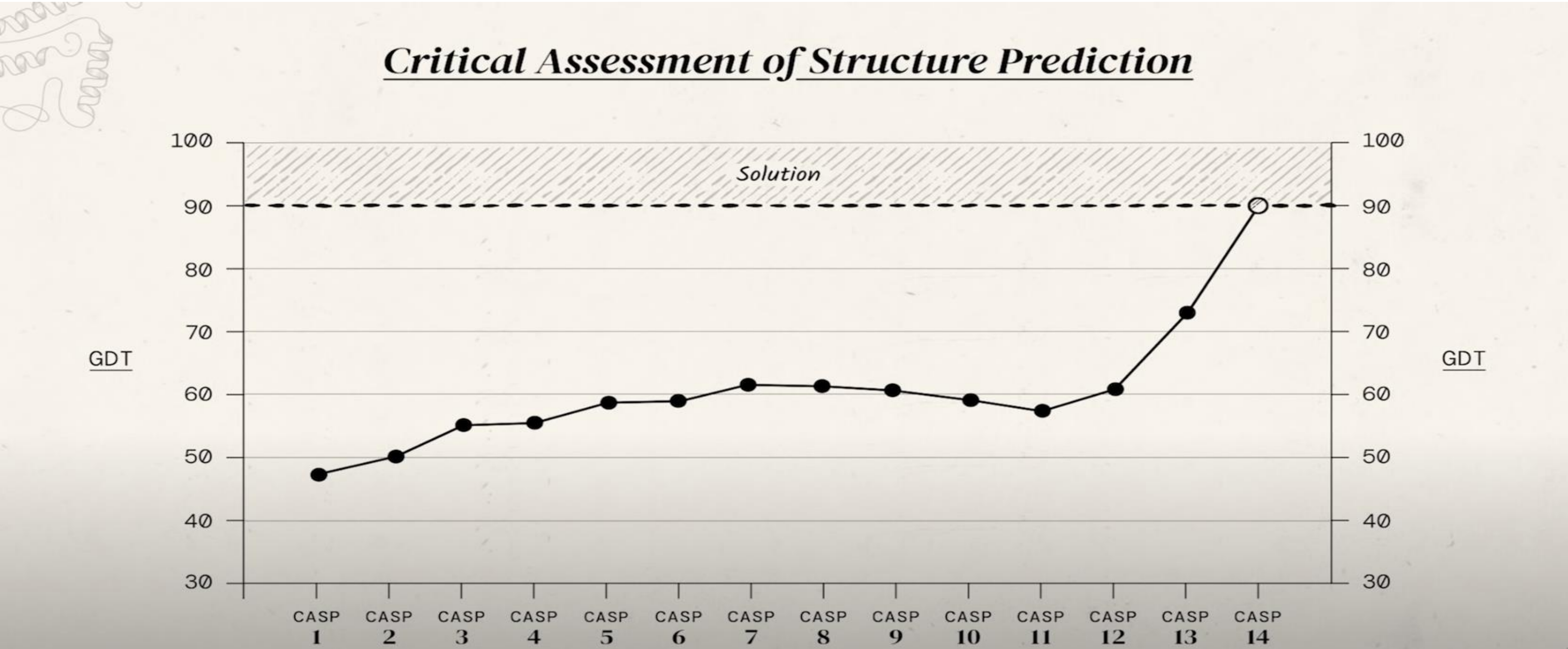
From: https://www.deepmind.com/



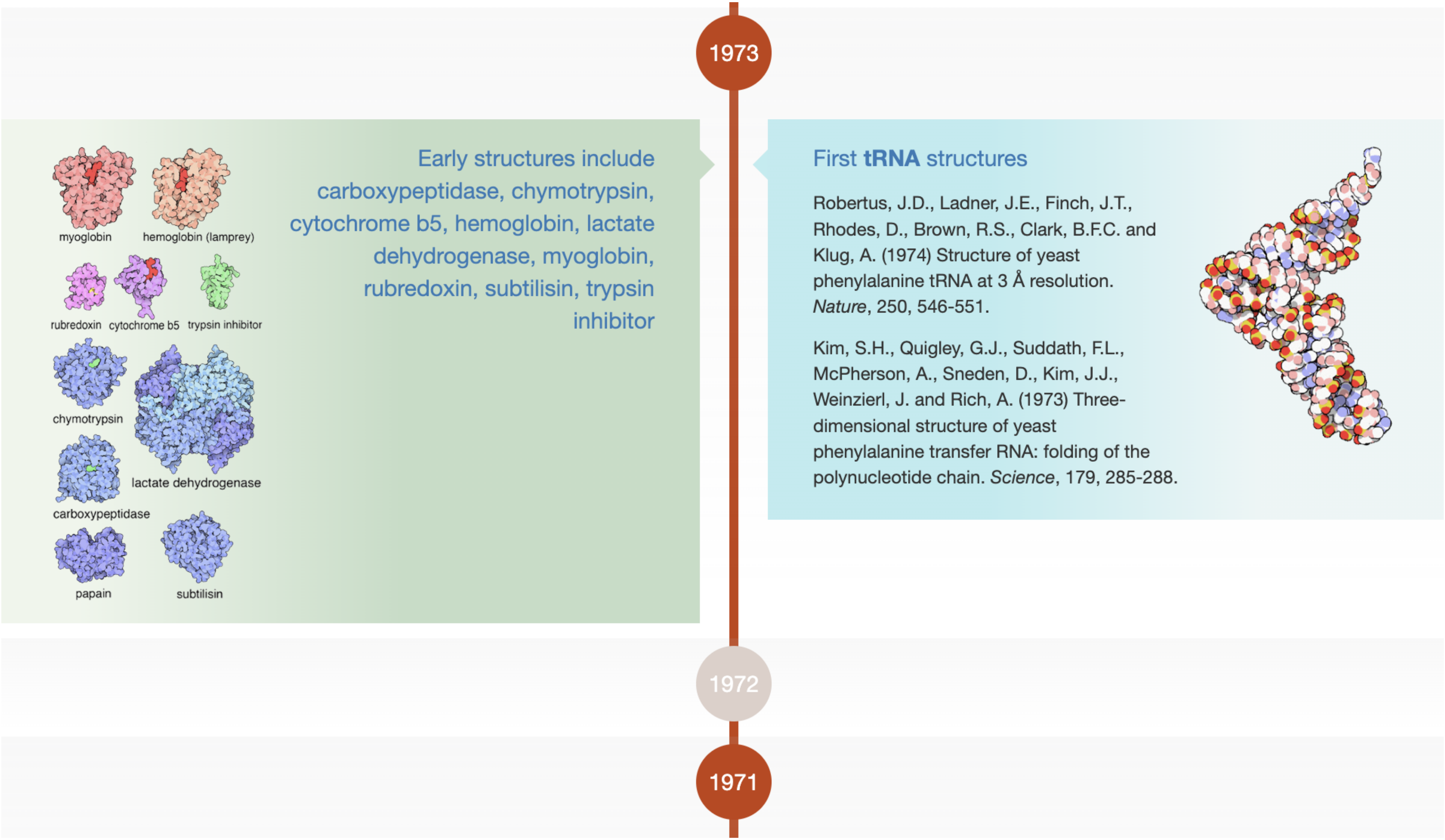


Global Distance Test

Measures of Progress in Inquiry: Protein folding and CASP (Jumper et al 2021)



https://www.rcsb.org/pages/about-us/history The Protein Data Bank established in 1971 with seven known structures



1973

Early structures include carboxypeptidase, chymotrypsin, cytochrome b5, hemoglobin, lactate dehydrogenase, myoglobin, rubredoxin, subtilisin, trypsin inhibitor

First **tRNA** structures

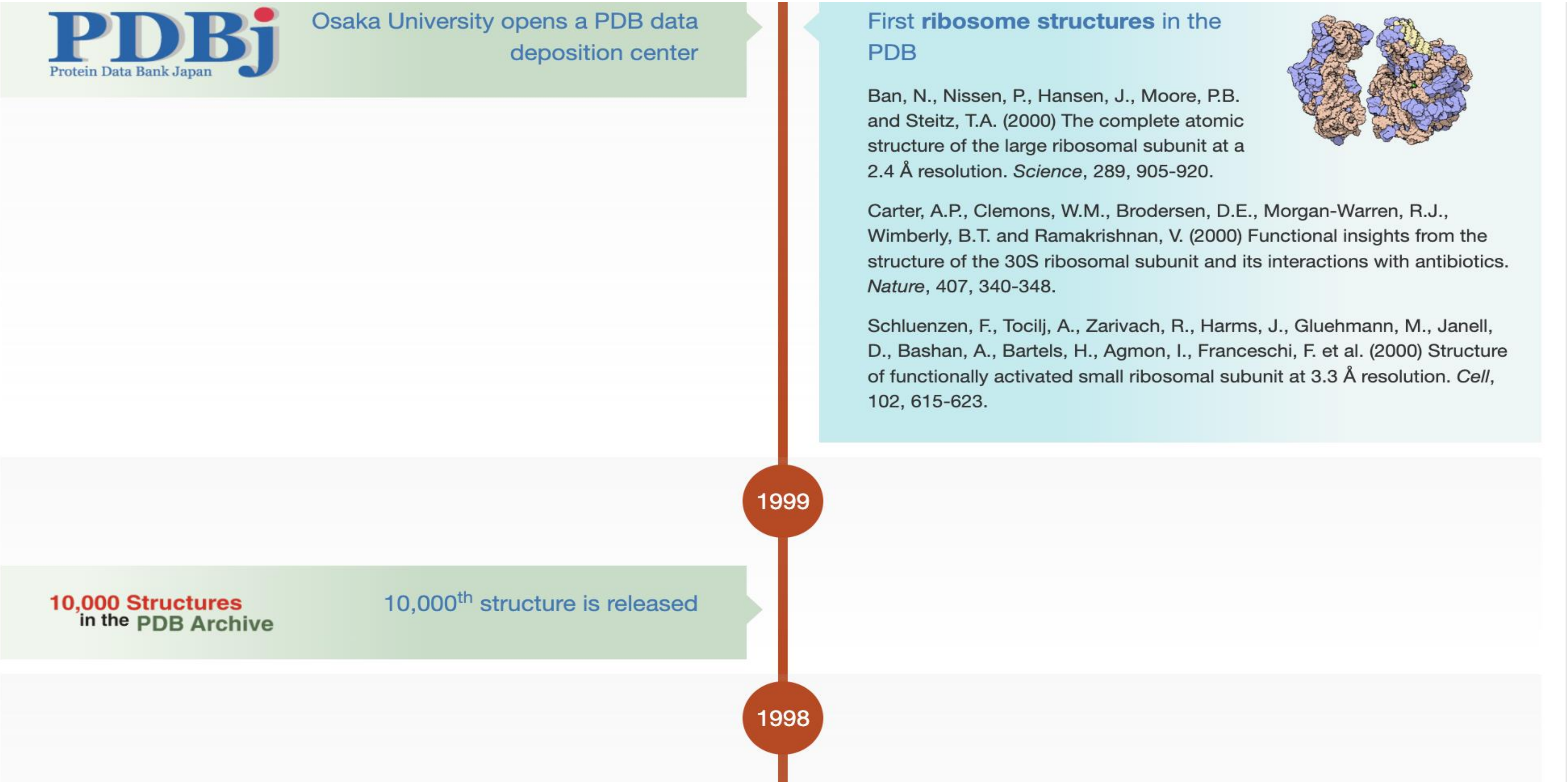
Robertus, J.D., Ladner, J.E., Finch, J.T., Rhodes, D., Brown, R.S., Clark, B.F.C. and Klug, A. (1974) Structure of yeast phenylalanine tRNA at 3 Å resolution. *Nature*, 250, 546-551.

Kim, S.H., Quigley, G.J., Suddath, F.L., McPherson, A., Sneden, D., Kim, J.J., Weinzierl, J. and Rich, A. (1973) Three-dimensional structure of yeast phenylalanine transfer RNA: folding of the polynucleotide chain. *Science*, 179, 285-288.

1972

1971

By 1999 the PDB contained 10,000 structures



Osaka University opens a PDB data deposition center

First **ribosome structures** in the PDB

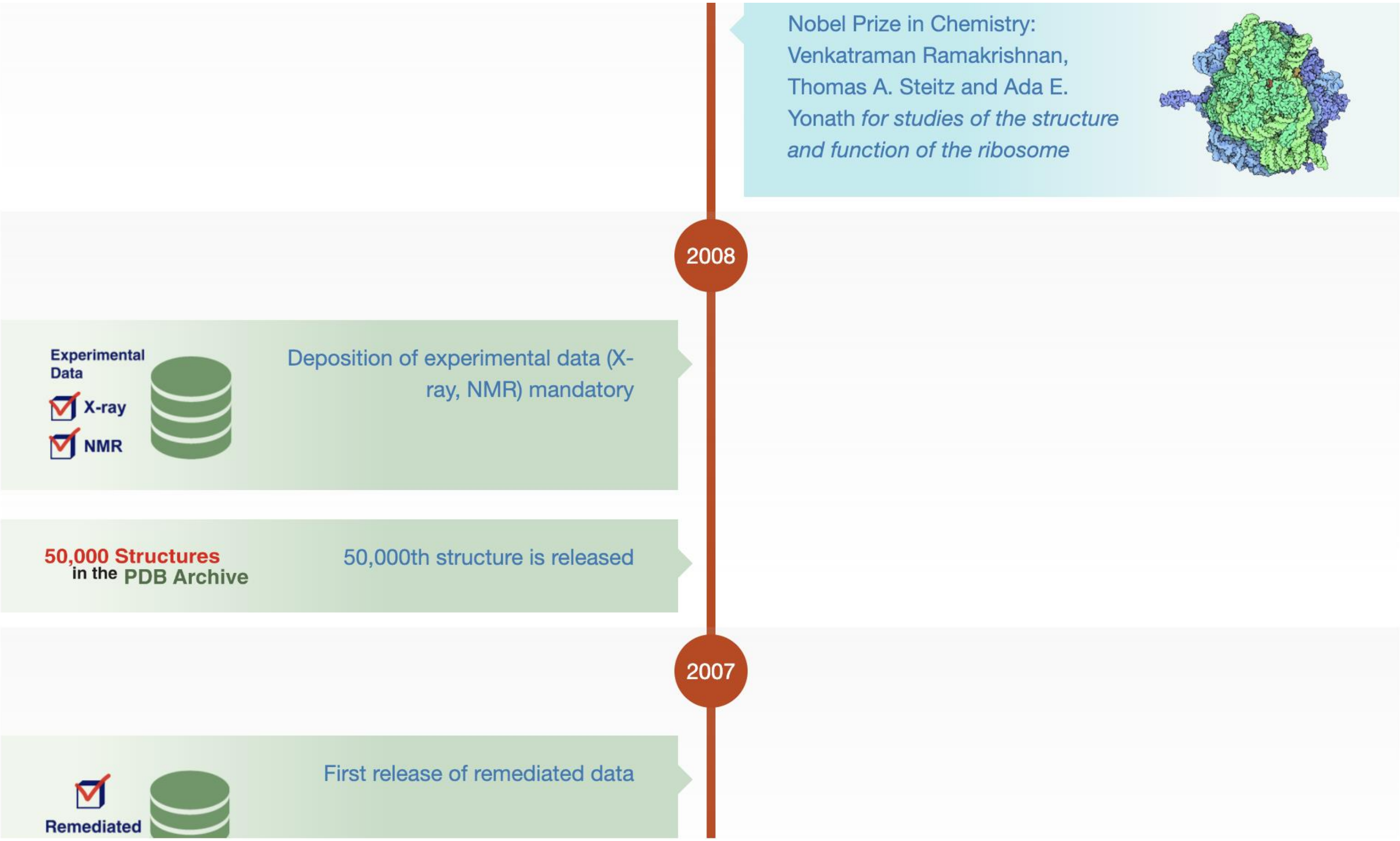
Ban, N., Nissen, P., Hansen, J., Moore, P.B. and Steitz, T.A. (2000) The complete atomic structure of the large ribosomal subunit at a 2.4 Å resolution. *Science*, 289, 905-920.

Carter, A.P., Clemons, W.M., Brodersen, D.E., Morgan-Warren, R.J., Wimberly, B.T. and Ramakrishnan, V. (2000) Functional insights from the structure of the 30S ribosomal subunit and its interactions with antibiotics. *Nature*, 407, 340-348.

Schluenzen, F., Tocilj, A., Zarivach, R., Harms, J., Gluehmann, M., Janell, D., Bashan, A., Bartels, H., Agmon, I., Franceschi, F. et al. (2000) Structure of functionally activated small ribosomal subunit at 3.3 Å resolution. *Cell*, 102, 615-623.

10,000th structure is released

By 2008, **50,000 structures**



Nobel Prize in Chemistry: Venkatraman Ramakrishnan, Thomas A. Steitz and Ada E. Yonath *for studies of the structure and function of the ribosome*

2008

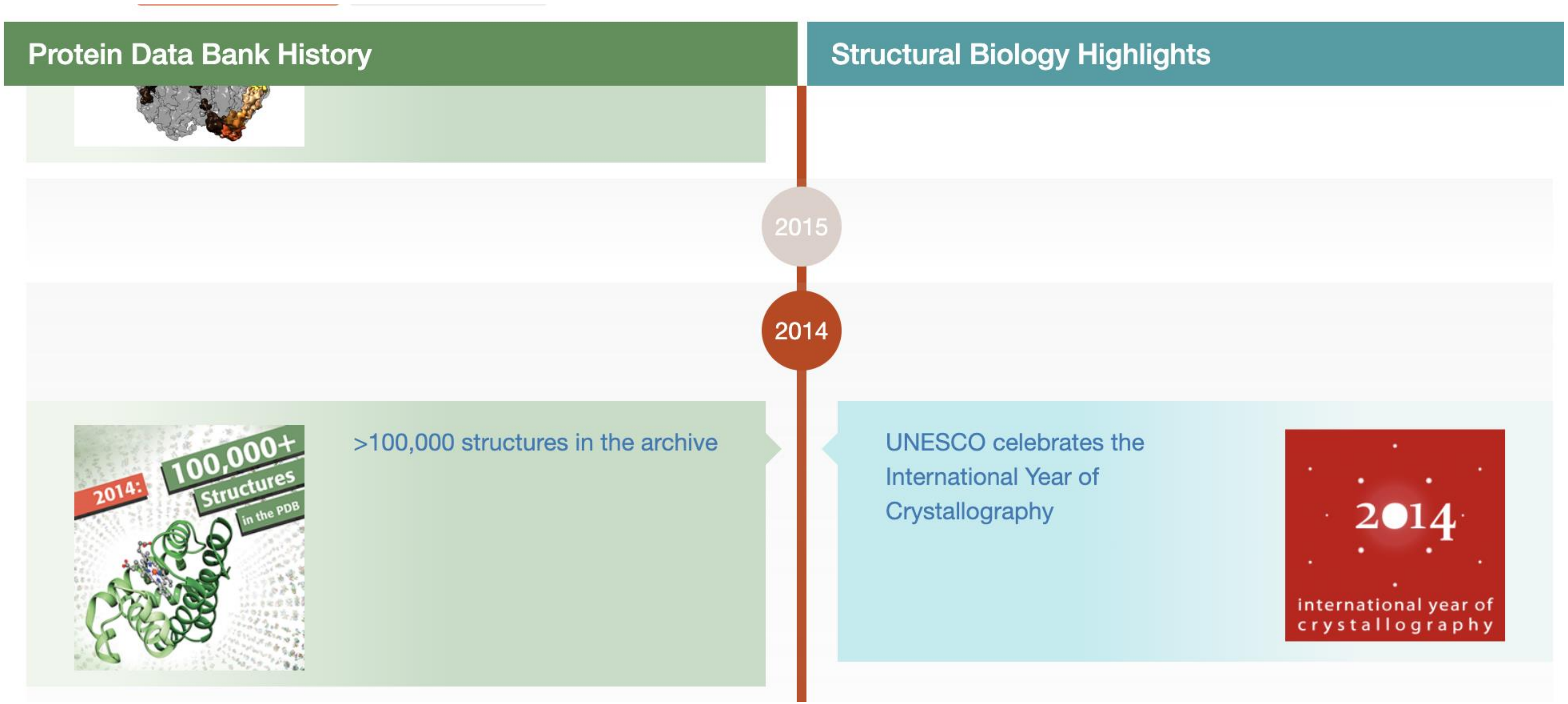
Deposition of experimental data (X-ray, NMR) mandatory

50,000th structure is released

2007

First release of remediated data

100,000 by 2014



Protein Data Bank History

Structural Biology Highlights

2015

2014

>100,000 structures in the archive

UNESCO celebrates the International Year of Crystallography

These 100,000 known structures served as the training set for early ML efforts.

- Training data is rich
- Optimization target for CASP competition is clearly defined (GDT)
- Even though aspects of the problem are known to be NP hard, constraints are relatively well-defined (albeit interacting in highly complex ways)

The Story

In 2022, a team led by David Baker at the University of Washington reported (Wicky et al. 2022; Dauparas et al. 2022) that, assisted by ML, they could design a wide variety of novel proteins in seconds instead of months to years.

The team's early approach to that problem was a two-phase process. In the first phase, researchers conceived a shape for a novel protein – often cast in terms of descriptions of short amide sequences from existing proteins. In the second phase, an ML program called Rosetta (developed by the team) then attempted to infer a sequence of amides that would spontaneously fold to the desired shape.

These first efforts produced few usable results. Typically, the amide sequences predicted by Rosetta did not fold into the desired 3D shape.

Thus, another step was required to ensure that the sequences folded as desired.

Simulating folding

One possible approach would be to simulate all the ways in which a given sequence of amides could fold.

The computational cost of that approach is intractably large.

Baker and colleagues accordingly took a different tack on the problem. They fed random amide sequences into a ML-based (3D-)structure-prediction neural network trained on the PDB.

This approach made predicted structures that looked increasingly like the desired structures. Those results appeared promising: Using this technique, the team created more than 100 small proteins (Anishchenko, Pellock, Chidyausiku *et al*. 2021), about 20% of which had structures that, to varying degrees, resembled the desired shape.

Although most of early protein-structure ML tools were trained to predict structures of individual amide chains, researchers soon discovered that such tools could be used to model assemblies of multiple interacting proteins.

Multiple-interacting-protein regimes, provide constraints that go beyond the single-protein cases, motivating Baker et al. to wonder whether a hallucinated multiple interacting-proteins approach would produce the desired results.

Alas, when they attempted to use genetically-modified microorganisms to produce the proteins predicted by this approach, none of their 150 designs folded as desired.

-- The system has being applied beyond its domain!

It was evident that the protein-folding solution space required even more domain specific constraints.

- Most members of the Baker team focused on the problem of predicting 3D structure of a protein, given its amide sequence.

- But one member of Baker's team, Justas Dauparas, was working on the inverse of that problem – given a 3D protein structure, determine its underlying amide sequence (Dauparas 2022 et al. 2022).
  - This effort produced a ML tool, ProteinMPNN, that can be used to check the results of ML programs that predict 3D protein structure, given a primary amide sequence.

- Baker and his team applied ProteinMPNN to the set of structures predicted by the hallucination technique and got stunning improvement in their results: 27 of 30 of the *hallucinated* designs that passed the ProteinMPNN check were confirmed by various experimental techniques.

In July 2022, Baker’s team described a pair of MLS methods that allow researchers to embed a specific sequence or structure in a novel protein (Wang et al. 2022). Using these approaches, the team has designed

1. proteins (enzymes) that catalyze specific reactions
2. proteins capable of binding to other molecules
3. a protein that could be used in a vaccine against a respiratory virus that is a leading cause of infant hospitalizations.

In addition to the MLS tools developed by the Baker team , some 40 MLS protein-design tools have been developed by other teams in recent years (Ferruz et al. 2022). These tools take a variety of approaches, including

a. *Fixed-backbone design.* Given a predetermined protein structure, an MLS determines an amide sequence for that protein.
b. *Structure generation.* An MLS trained on protein structures generates novel protein structures. In this approach, the control of the output is often limited.
c. *Sequence generation.* Using language models, An MLS learns to 'speak' protein. These tools can be fine-tuned to generate novel sequences resembling members of specific protein families.
d. *Sequence and structure design*. One of the techniques in this class, in an approach called *inpainting,* researchers input a structure or sequence that they want included in a protein, and an MLS fills in the rest.

Many of these tools tackle the inverse folding problem. Some are based on an architecture similar to that of GPT-3 (Brown et al. 2020), a general-purpose text-oriented neural network.

Last year, London-based DeepMind spun off Isomorphic Labs (Isomorphic Laboratories 2022), a company that will apply MLS tools such as DeepMind's AlphaFold (DeepMind 2022) to protein-based-drug discovery.

What should we take away from the protein folding application of ML

- The protein-design example strongly suggests that to be useful, ML must incorporate
    - a range of topic-specific constraints
    - domain specific knowledge
    - training sets that embody deep human-level expertise/experimental results

- The protein folding case also demonstrates the ways that researchers can usefully constrain the hallucinating ML system, shaping results and guiding the system towards what matters.

What do our reflections on NFL and Alphfold imply about the role of ML in progress in scientific inquiry?

Is there some sense in which inquiry can be said to make “progress” through “iteration”.   That question, and answers to it, clearly depends on what we mean by “inquiry”, “progress”, and “iteration” and how they are related.

The problem of progress in inquiry.

Peirce, for example, held that inquiry, through iteration, converges on truth - the *objective* of inquiry

Regardless of whether we think Peirce's view is defensible, however, it does highlight two important aspects of the traditional conception of inquiry:

(a) in some sense ordinary scientific inquiry is iterative, and

(b) inquiry has an objective.

Generally, we have been interested in optimizing on the objective of inquiry if possible.

“Inquiry properly carried on will reach some definite and fixed result or approximate indefinitely toward that limit”

[Peirce 1931, vol.1 458]

“It is unphilosophical to suppose that, with regard to any given question, (which has any clear meaning), investigation would not bring forth a solution of it, if it were carried far enough."

[Peirce 1965,268-9]

“There is a fundamental assumption or thesis in philosophy which says that scientific knowledge may be characterized by convergence to a correct hypothesis in the limit of empirical scientific inquiry"

[Hendricks 2001, 1]

Algorithm-based inquiry is often considered the paradigm of rational inquiry because at first glance it appears to be mechanical, reproducible, incremental, and finite. We know many examples of algorithms. In particular, the rules for computing basic arithmetic operations such as multiplication or division, and the rules for bisecting an angle using compass and straightedge, and Euclid's algorithm for calculating the greatest common divisor, are all algorithms.

The phrase, "optimizing on an objective" is also vague in various ways.   One way of reducing that vagueness is to render "optimizing on an objective" in mathematically well-defined language. The canonical way of doing this is to define "optimizing on an objective" as "maximizing or minimizing the value of a function which we regard as measuring how well an objective of interest is attained.   A function that measures (in the sense of Halmos 1950) how well some objective of interest is attained is called an *objective function*.

Algorithmic learning theory (ALT)

Topics for ALT include prominently:

Turing Learning:

What can a Turing machine learn?

Most prominently: Gold’s “language learning in the limit”

Efficiency:

Convergence to the truth in the smallest number of data points

Revision:

Convergence to the truth with the smallest number of revisions to hypotheses

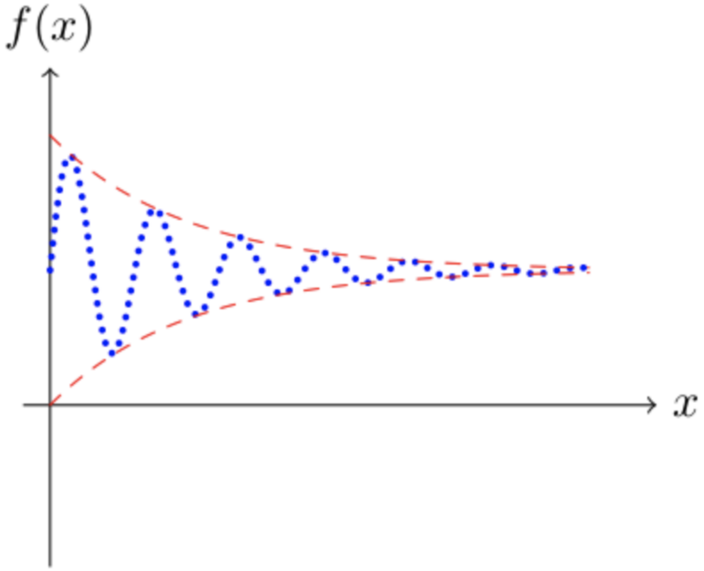
Peirce

Peirce understood the iterative process of inquiry to involve something like an algorithm, a self-correcting convergent process.

Truth is understood to be that to which science converges in the limit of inquiry.

But:

We already have many truths (we don’t have to wait).
The converging process of inquiry may miss buried treasure.



Peirce’s model of inquiry assumes an end (in the limit) but is less clear about the beginning.

In practice, human inquiry usually begins from somewhere.

A concern that defines a domain

Inquiry also tends to generate new questions and opens new possibilities rather than providing improved answers to the same question via an iterative process.

This is not to deny that if a domain can be constrained in suitable ways it can become a fertile problem space for ML. It can become a region suitable for methods that converge towards well-defined goals (optimization).

Open Questions and Partial Conclusions

How do we understand the definition of *domain* here, and what does this tell us about inquiry?
Can it be formally characterized?

Domain is a region of concern. Human beings have concerns shaped partially by our resource constraints and goals shaped by our values.

Not everything is relevant for us and not every learning problem is equally likely.

THANK YOU!