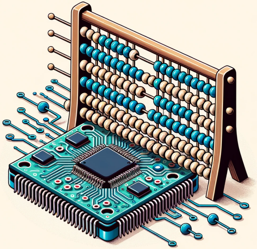
The Role of Machine Learning for Mathematics

What impact can we expect ML to have on theoretical math research?

Christoph Spiegel

Friday, 19th of September 2024

Zuse Institute Berlin



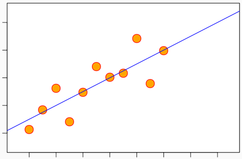
Three Views of Machine Learning

1. **Approximation**
2. **Generalization**
3. **Emergence**
4. Pointers for your own research

Approximation

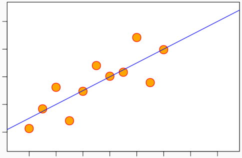
Approximation

Given $A \in \mathbb{R}^{m \times n}$ and $\mathbf{b} \in \mathbb{R}^m$, the map $f : \mathbb{R}^n \to \mathbb{R}^m$, $\mathbf{x} \mapsto A\,\mathbf{x} + \mathbf{b}$ is an affine linear map.

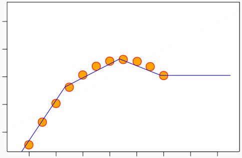


Approximation

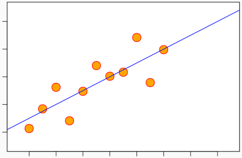
Given $A \in \mathbb{R}^{m\times n}$ and $\mathbf{b} \in \mathbb{R}^m$, the map $f : \mathbb{R}^n \to \mathbb{R}^m$, $\mathbf{x} \mapsto A\mathbf{x} + \mathbf{b}$ is an affine linear map.



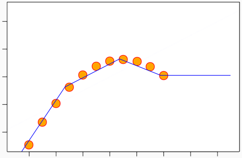
An ℓ-layer perceptron is a map $\mathbf{x} \mapsto f_\ell \circ g \ldots \circ f_2 \circ g \circ f_1(\mathbf{x})$ where $f_1, \ldots, f_\ell$ are affine linear maps and g a *non-linear* activation function.



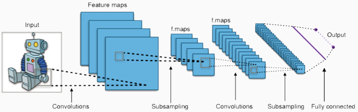
Approximation



Given $A \in \mathbb{R}^{m\times n}$ and $\mathbf{b} \in \mathbb{R}^m$, the map $f : \mathbb{R}^n \to \mathbb{R}^m$, $\mathbf{x} \mapsto A\mathbf{x} + \mathbf{b}$ is an affine linear map.



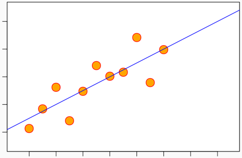
An ℓ-layer perceptron is a map $\mathbf{x} \mapsto f_\ell \circ g \ldots \circ f_2 \circ g \circ f_1(\mathbf{x})$ where $f_1, \ldots, f_\ell$ are affine linear maps and g a *non-linear* activation function.



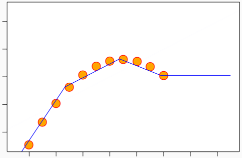
A neural network is a multilayer perceptron that accounts for some structure in the input.

Approximation

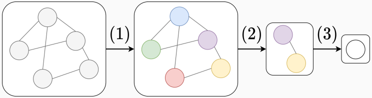
Given $A \in \mathbb{R}^{m \times n}$ and $\mathbf{b} \in \mathbb{R}^m$, the map $f : \mathbb{R}^n \to \mathbb{R}^m$, $\mathbf{x} \mapsto A\mathbf{x} + \mathbf{b}$ is an affine linear map.



An ℓ-layer perceptron is a map $\mathbf{x} \mapsto f_\ell \circ g \ldots \circ f_2 \circ g \circ f_1(\mathbf{x})$ where $f_1, \ldots, f_\ell$ are affine linear maps and g a *non-linear* activation function.

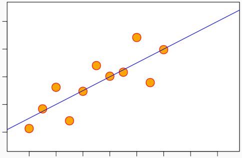


A neural network is a multilayer perceptron that accounts for some structure in the input.

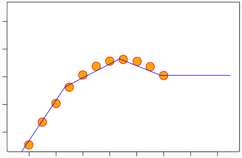


Approximation

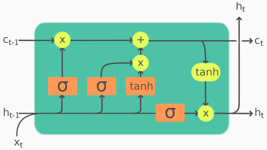
Given $A \in \mathbb{R}^{m \times n}$ and $\mathbf{b} \in \mathbb{R}^m$, the map $f : \mathbb{R}^n \to \mathbb{R}^m$, $\mathbf{x} \mapsto A\mathbf{x} + \mathbf{b}$ is an affine linear map.



An ℓ-layer perceptron is a map $\mathbf{x} \mapsto f_\ell \circ g \ldots \circ f_2 \circ g \circ f_1(\mathbf{x})$ where $f_1, \ldots, f_\ell$ are affine linear maps and g a *non-linear* activation function.



A neural network is a multilayer perceptron that accounts for some structure in the input.



Approximation

Let us assume we are using ReLU activation, that is $g(x) = \max(x, 0)$.

Universal Approximation Theorem

Any continuous function with compact support can be approximated arbitrarily closely by a 2-layer perceptron.

Approximation

Let us assume we are using ReLU activation, that is $g(x) = \max(x, 0)$.

Universal Approximation Theorem

Any continuous function with compact support can be approximated arbitrarily closely by a 2-layer perceptron.

Under additional assumptions it can also be approximated by a (not necessarily 2-layer) perceptron with width at most dimension of the input.

See Shen et al. [15] for a study of the optimal depth–width–tradeoff.

Approximation

Let us assume we are using ReLU activation, that is $g(x) = \max(x, 0)$.

Universal Approximation Theorem

Any continuous function with compact support can be approximated arbitrarily closely by a 2-layer perceptron.

Under additional assumptions it can also be approximated by a (not necessarily 2-layer) perceptron with width at most dimension of the input.

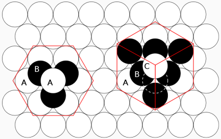
See Shen et al. [15] for a study of the optimal depth–width–tradeoff.

Question

How few layers can a multilayer perceptron have that *exactly* models a specific function, e.g., the maximum of its input? See Hertrich et al. [9]

Approximation

The answer to many questions is given by concrete objects …



Sphere packing

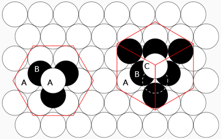
What is the densest arrangement of congruent spheres in Euclidean space?

Approximation

The answer to many questions is given by concrete objects …

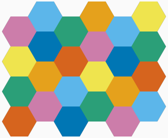
Sphere packing

What is the densest arrangement of congruent spheres in Euclidean space?



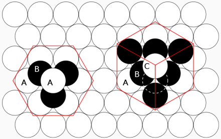
Hadwiger Nelson

How many colors do you need such that no two points at unit distance have the same color?



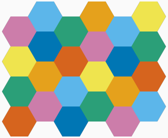
Approximation

The answer to many questions is given by concrete objects ...



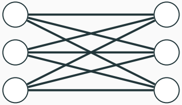
Sphere packing

What is the densest arrangement of congruent spheres in Euclidean space?



Hadwiger Nelson

How many colors do you need such that no two points at unit distance have the same color?



Mantel's Theorem

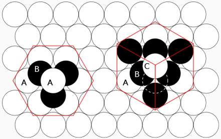
What is the maximum edge density of a graph without triangles?

Approximation

The answer to many questions is given by concrete objects …

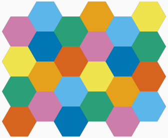
Sphere packing

What is the densest arrangement of congruent spheres in Euclidean space?



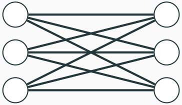
Hadwiger Nelson

How many colors do you need such that no two points at unit distance have the same color?



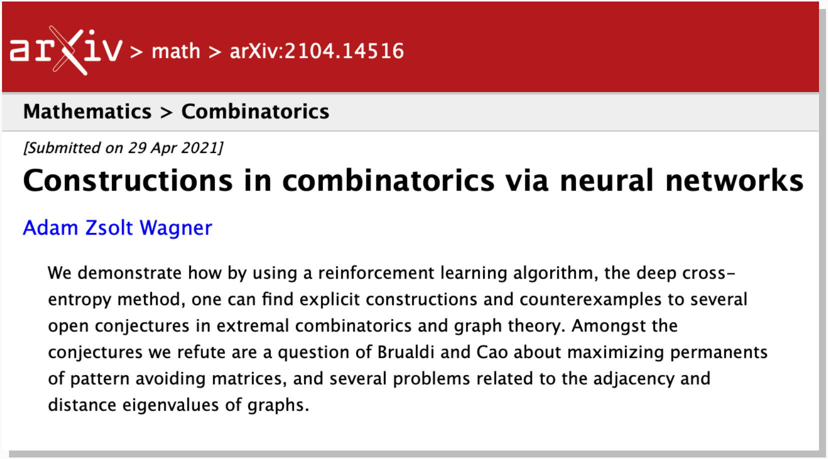
Mantel's Theorem

What is the maximum edge density of a graph without triangles?



Idea. *Somehow* get a neural network to represent these objects.

Approximation



arXiv > math > arXiv:2104.14516

Mathematics > Combinatorics

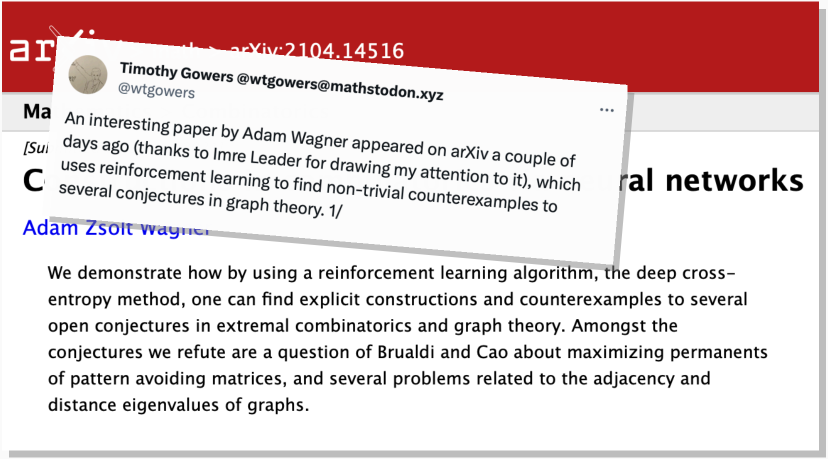
[Submitted on 29 Apr 2021]

Constructions in combinatorics via neural networks

Adam Zsolt Wagner

We demonstrate how by using a reinforcement learning algorithm, the deep cross-entropy method, one can find explicit constructions and counterexamples to several open conjectures in extremal combinatorics and graph theory. Amongst the conjectures we refute are a question of Brualdi and Cao about maximizing permanents of pattern avoiding matrices, and several problems related to the adjacency and distance eigenvalues of graphs.

Approximation



arXiv:2104.14516

Timothy Gowers @wtgowers@mathstodon.xyz
@wtgowers

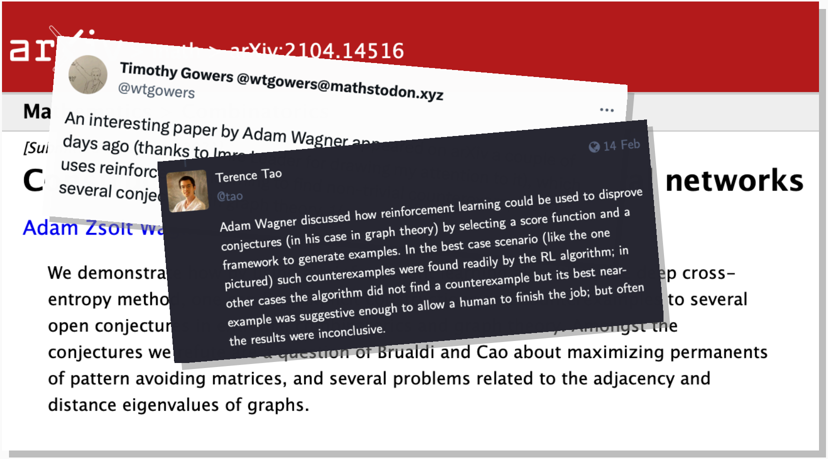
An interesting paper by Adam Wagner appeared on arXiv a couple of days ago (thanks to Imre Leader for drawing my attention to it), which uses reinforcement learning to find non-trivial counterexamples to several conjectures in graph theory. 1/

ral networks

Adam Zsolt Wagner

We demonstrate how by using a reinforcement learning algorithm, the deep cross-entropy method, one can find explicit constructions and counterexamples to several open conjectures in extremal combinatorics and graph theory. Amongst the conjectures we refute are a question of Brualdi and Cao about maximizing permanents of pattern avoiding matrices, and several problems related to the adjacency and distance eigenvalues of graphs.

Approximation



arXiv > math > arXiv:2104.14516

Ma

[Su

C … networks

Adam Zsolt Wa

We demonstra … ep cross–entropy metho … to several open conjectur … Amongst the conjectures we … a question of Brualdi and Cao about maximizing permanents of pattern avoiding matrices, and several problems related to the adjacency and distance eigenvalues of graphs.

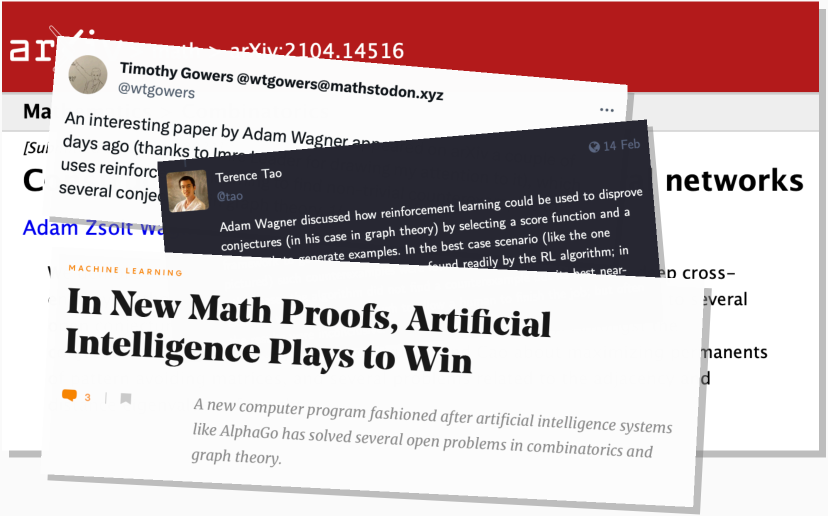
Timothy Gowers @wtgowers@mathstodon.xyz
@wtgowers

An interesting paper by Adam Wagner app
days ago (thanks to Imre
uses reinforc
several conje

Terence Tao
@tao
14 Feb

Adam Wagner discussed how reinforcement learning could be used to disprove conjectures (in his case in graph theory) by selecting a score function and a framework to generate examples. In the best case scenario (like the one pictured) such counterexamples were found readily by the RL algorithm; in other cases the algorithm did not find a counterexample but its best near-example was suggestive enough to allow a human to finish the job; but often the results were inconclusive.

Approximation



math > arXiv:2104.14516

Ma

[Su

C networks

Adam Zsolt Wa

p cross-

everal

manents

d

Timothy Gowers @wtgowers@mathstodon.xyz
@wtgowers

An interesting paper by Adam Wagner ap
days ago (thanks to Imre
uses reinforc
several conje

Terence Tao
@tao
14 Feb

Adam Wagner discussed how reinforcement learning could be used to disprove conjectures (in his case in graph theory) by selecting a score function and a
generate examples. In the best case scenario (like the one
found readily by the RL algorithm; in
best near-

MACHINE LEARNING

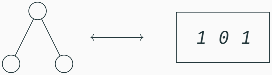
In New Math Proofs, Artificial Intelligence Plays to Win

3

A new computer program fashioned after artificial intelligence systems like AlphaGo has solved several open problems in combinatorics and graph theory.

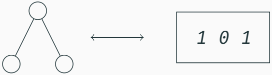
Approximation

1. Represent discrete objects such as graphs or 0-1-matrices through **binary vectors** and define a **cost function**.

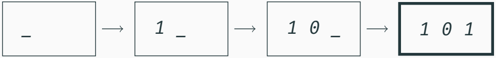


Approximation

1. Represent discrete objects such as graphs or 0-1-matrices through **binary vectors** and define a **cost function**.

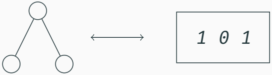


2. Model a distribution over these vectors through a **neural network evaluated in a round-based fashion**.

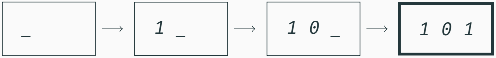


Approximation

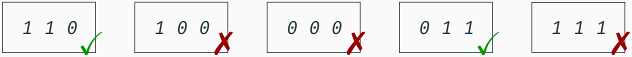
1. Represent discrete objects such as graphs or 0-1-matrices through **binary vectors** and define a **cost function**.



2. Model a distribution over these vectors through a **neural network evaluated in a round-based fashion**.



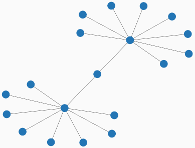
3. Update the parameters of the neural network through **Reinforcement Learning** to incentivise lower cost.



Approximation

Any connected graph on n vertices with matching number μ and largest eigenvalue λ_1 satisfies $\lambda_1 + \mu \geq \sqrt{n-1} + 1$.

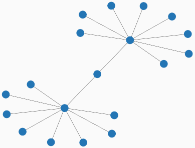
Aouchiche and Hansen [1]



Approximation

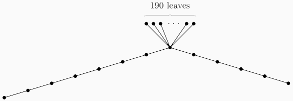
Any connected graph on n vertices with matching number μ and largest eigenvalue λ_1 satisfies $\lambda_1 + \mu \geq \sqrt{n-1} + 1$.

Aouchiche and Hansen [1]



Any connected graph with diameter D, proximity π and distance spectrum $\partial_1 \geq \ldots \geq \partial_n$ satisfies $\pi + \partial_{\lfloor 2D/3 \rfloor} > 0$.

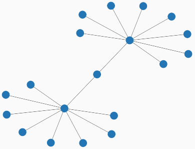
Aouchiche and Hansen [2]



Approximation

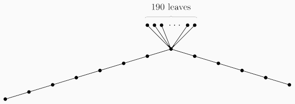
Any connected graph on n vertices with matching number μ and largest eigenvalue λ_1 satisfies $\lambda_1 + \mu \geq \sqrt{n-1} + 1$.

Aouchiche and Hansen [1]



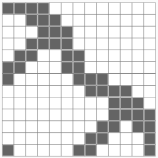
Any connected graph with diameter D, proximity π and distance spectrum $\partial_1 \geq \ldots \geq \partial_n$ satisfies $\pi + \partial_{\lfloor 2D/3 \rfloor} > 0$.

Aouchiche and Hansen [2]

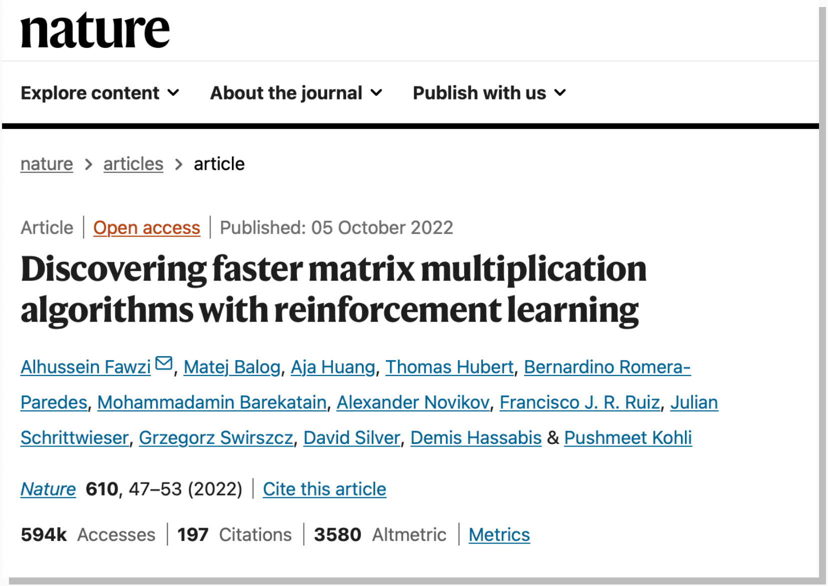


What is the largest permanent of an $n \times n$ 0-1 matrix $A = (a_{i,j})$ that avoids the pattern $a_{i_1,i_3} = a_{i_2,i_1} = a_{i_3,i_2} = 1$?

Brualdi and Cao [3]



Approximation



nature

Explore content ⌄ About the journal ⌄ Publish with us ⌄

nature › articles › article

Article | Open access | Published: 05 October 2022

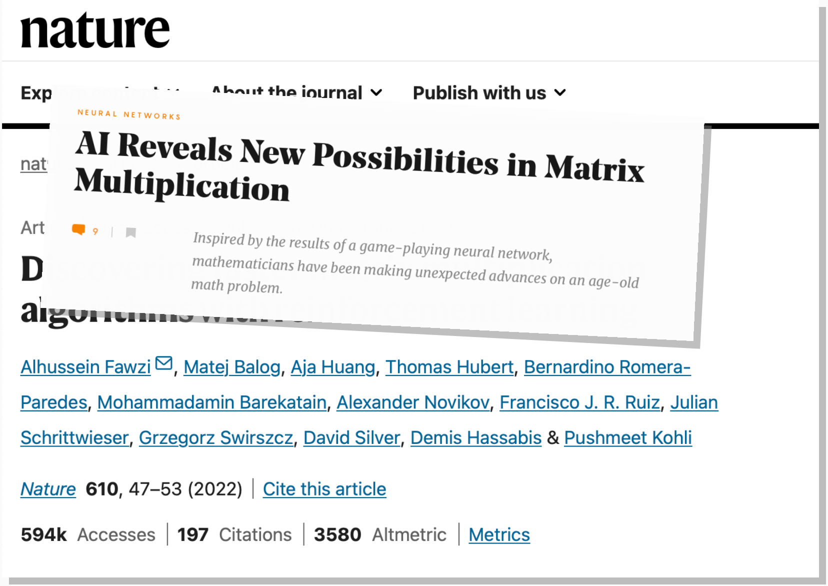
Discovering faster matrix multiplication algorithms with reinforcement learning

Alhussein Fawzi ✉, Matej Balog, Aja Huang, Thomas Hubert, Bernardino Romera-Paredes, Mohammadamin Barekatain, Alexander Novikov, Francisco J. R. Ruiz, Julian Schrittwieser, Grzegorz Swirszcz, David Silver, Demis Hassabis & Pushmeet Kohli

Nature **610**, 47–53 (2022) | Cite this article

594k Accesses | **197** Citations | **3580** Altmetric | Metrics

Approximation



nature

Exp · About the journal · Publish with us

nat

Art

D

al

NEURAL NETWORKS

AI Reveals New Possibilities in Matrix Multiplication

9

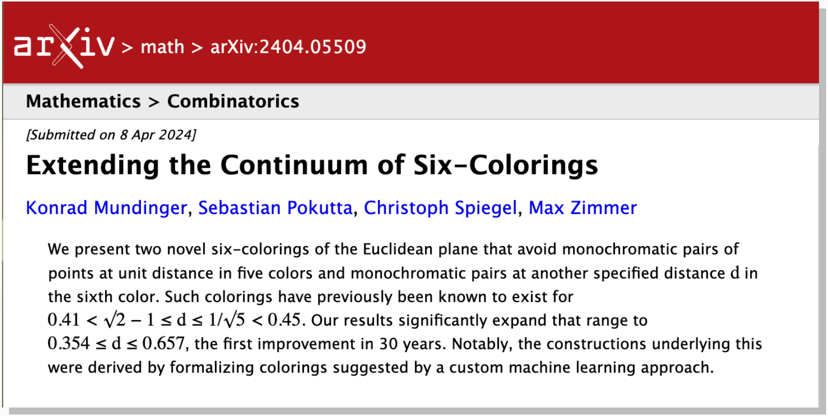
Inspired by the results of a game-playing neural network, mathematicians have been making unexpected advances on an age-old math problem.

Alhussein Fawzi, Matej Balog, Aja Huang, Thomas Hubert, Bernardino Romera-Paredes, Mohammadamin Barekatain, Alexander Novikov, Francisco J. R. Ruiz, Julian Schrittwieser, Grzegorz Swirszcz, David Silver, Demis Hassabis & Pushmeet Kohli

Nature **610**, 47–53 (2022) | Cite this article

594k Accesses | **197** Citations | **3580** Altmetric | Metrics

Approximation



arXiv > math > arXiv:2404.05509

Mathematics > Combinatorics

[Submitted on 8 Apr 2024]

Extending the Continuum of Six-Colorings

Konrad Mundinger, Sebastian Pokutta, Christoph Spiegel, Max Zimmer

We present two novel six-colorings of the Euclidean plane that avoid monochromatic pairs of points at unit distance in five colors and monochromatic pairs at another specified distance d in the sixth color. Such colorings have previously been known to exist for $0.41 < \sqrt{2} - 1 \le d \le 1/\sqrt{5} < 0.45$. Our results significantly expand that range to $0.354 \le d \le 0.657$, the first improvement in 30 years. Notably, the constructions underlying this were derived by formalizing colorings suggested by a custom machine learning approach.

Approximation

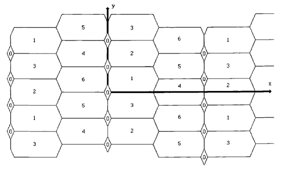
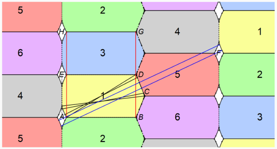
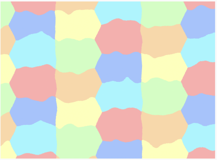


FIG. 3. A good 7-coloring of $(\mathbb{R}^2, 1)$.





Approximation

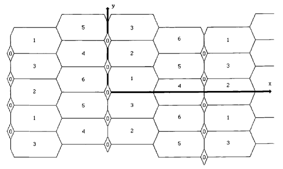
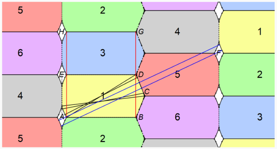
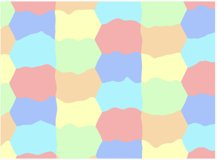
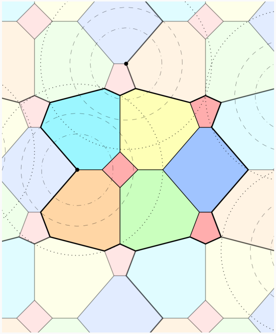
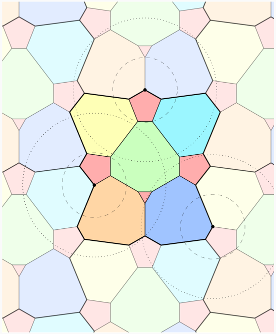


FIG. 3. A good 7-coloring of $(\mathbb{R}^2, 1)$.





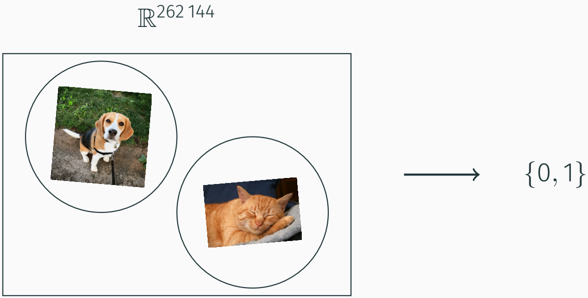




Generalization

Generalization

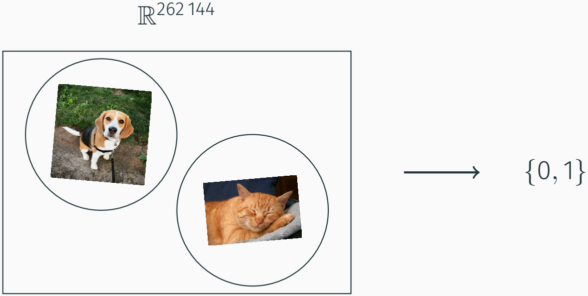
More commonly, machine learning applications are (or at least used to be) supervised **regression or classification tasks** on datasets:



$$\mathbb{R}^{262\,144} \longrightarrow \{0,1\}$$

Generalization

More commonly, machine learning applications are (or at least used to be) supervised **regression or classification tasks** on datasets:



Universal approximation tells us that we can fit *any* dataset, even random noise! Why would this generalize to any actual application?

Generalization

Classical statistics emphasizes simple models, guarantees, assumptions about the data, and interpretability of the output.

Generalization

Classical statistics emphasizes simple models, guarantees, assumptions about the data, and interpretability of the output.

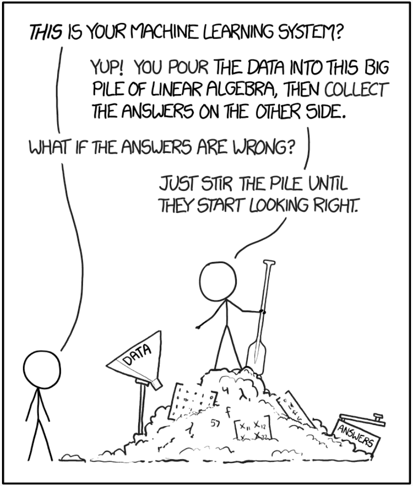
Big Data relies on overparameterization, offers little formal guarantees or interpretability, and accepts behavior like adversarial examples.

Generalization

Classical statistics emphasizes simple models, guarantees, assumptions about the data, and interpretability of the output.

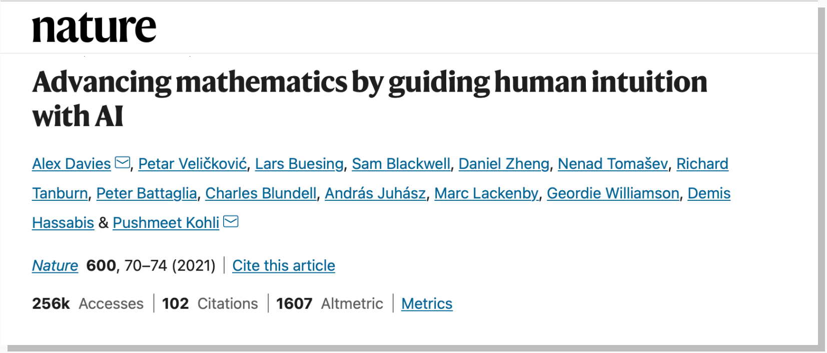
Big Data relies on overparameterization, offers little formal guarantees or interpretability, and accepts behavior like adversarial examples.

Generalization is achieved through many practical tools, but the current state of machine learning has been likened to **alchemy**. Ali Rahimi at NeurIPS 2017


THIS IS YOUR MACHINE LEARNING SYSTEM?
YUP! YOU POUR THE DATA INTO THIS BIG PILE OF LINEAR ALGEBRA, THEN COLLECT THE ANSWERS ON THE OTHER SIDE.
WHAT IF THE ANSWERS ARE WRONG?
JUST STIR THE PILE UNTIL THEY START LOOKING RIGHT.
DATA
ANSWERS

xkcd.com

Generalization



nature

Advancing mathematics by guiding human intuition with AI

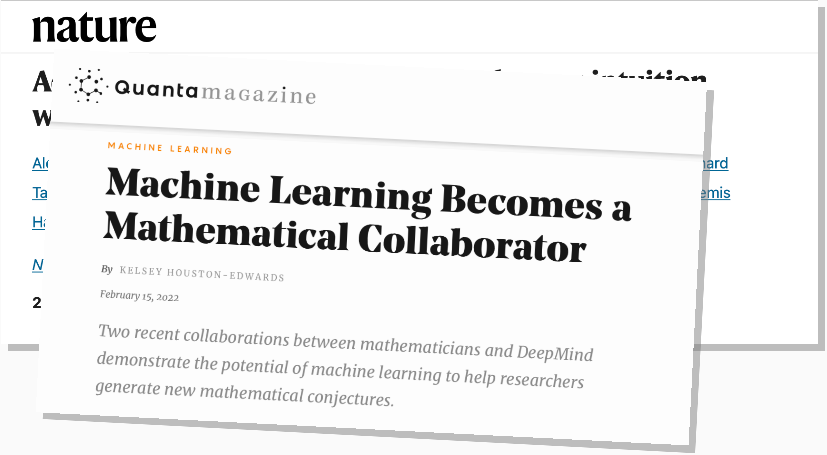
Alex Davies, Petar Veličković, Lars Buesing, Sam Blackwell, Daniel Zheng, Nenad Tomašev, Richard Tanburn, Peter Battaglia, Charles Blundell, András Juhász, Marc Lackenby, Geordie Williamson, Demis Hassabis & Pushmeet Kohli

Nature **600**, 70–74 (2021) | Cite this article

256k Accesses | **102** Citations | **1607** Altmetric | Metrics

Two collaborations from Google DeepMind with András Juhász and Marc Lackenby (knot theory) and Geordie Williamson (representation theory).

Generalization



nature

A
W
Al
Ta
Ha
N
2

intuition
ard
emis

Quantamagazine

MACHINE LEARNING

Machine Learning Becomes a Mathematical Collaborator

By KELSEY HOUSTON-EDWARDS

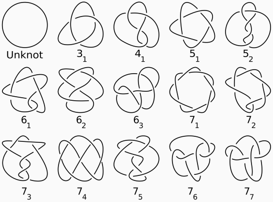
February 15, 2022

Two recent collaborations between mathematicians and DeepMind demonstrate the potential of machine learning to help researchers generate new mathematical conjectures.

Two collaborations from Google DeepMind with András Juhász and Marc Lackenby (knot theory) and Geordie Williamson (representation theory).

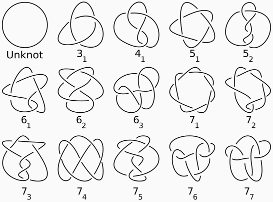
Generalization

A knot is an embedding of the S^1 into $\mathbb{R}^3$. Two knots equivalent if they can continuously be deformed into each other. An invariant is a function on equivalence classes of knots.



Generalization

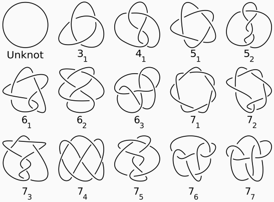
A knot is an embedding of the S^1 into $\mathbb{R}^3$. Two knots equivalent if they can continuously be deformed into each other. An invariant is a function on equivalence classes of knots.



Juhász and Lackenby believed in an undiscovered relation between **geometric** and **algebraic** varieties.

Generalization

A knot is an embedding of the S^1 into $\mathbb{R}^3$. Two knots equivalent if they can continuously be deformed into each other. An invariant is a function on equivalence classes of knots.



Juhász and Lackenby believed in an undiscovered relation between **geometric** and **algebraic** varieties.

geometric varieties

Meridional translation
Longitudinal translation
Short geodesic
Injectivity radius
…

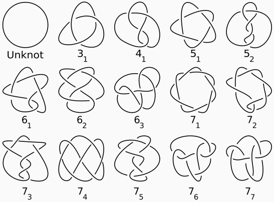
$\longrightarrow$

algebraic varieties

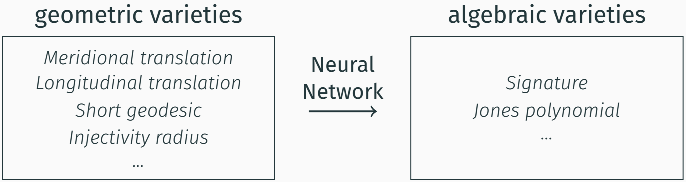
Signature
Jones polynomial
…

Generalization

A knot is an embedding of the S^1 into $\mathbb{R}^3$. Two knots equivalent if they can continuously be deformed into each other. An invariant is a function on equivalence classes of knots.



Juhász and Lackenby believed in an undiscovered relation between **geometric** and **algebraic** varieties.



geometric varieties

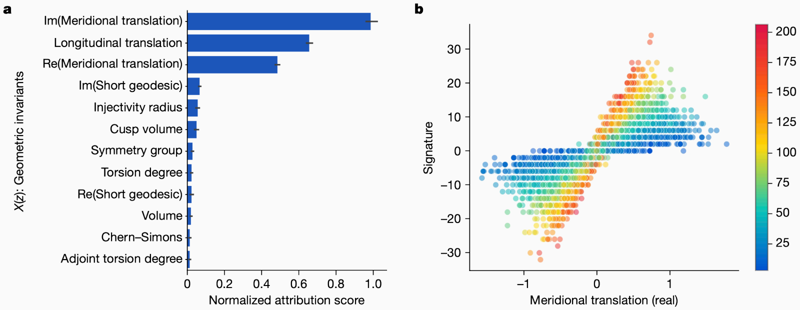
Meridional translation
Longitudinal translation
Short geodesic
Injectivity radius
...

Neural Network →

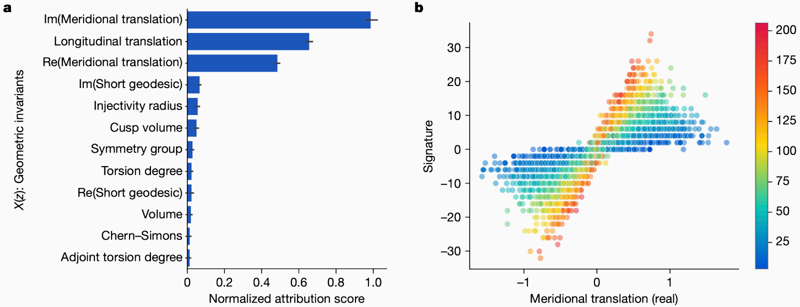
algebraic varieties

Signature
Jones polynomial
...

Generalization



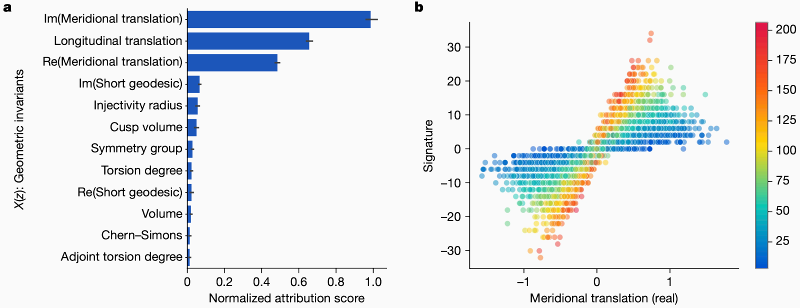
Generalization



Based on this, it was conjectured that

$$\left|\, 2\sigma(K) - \mathrm{Re}\left(\lambda(K)/\mu(K)\right) \right| < c_1 \,\mathrm{vol}(K) + c_2$$

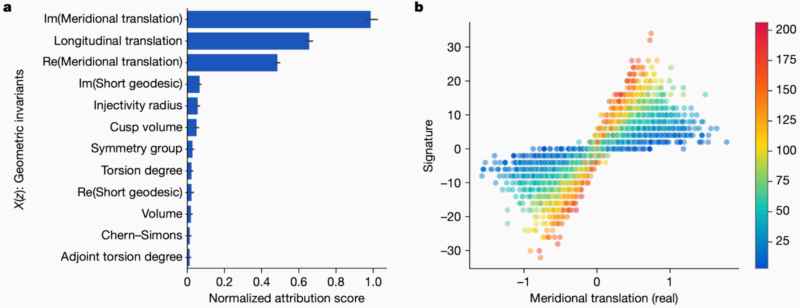
Generalization



Based on this, it was conjectured that

$$\cancel{\left|2\sigma(K) - \mathrm{Re}\left(\lambda(K)/\mu(K)\right)\right| < c_1 \mathrm{vol}(K) + c_2}$$

Generalization



Based on this, it was conjectured that

$$~~|2\sigma(K) - \mathrm{Re}\,(\lambda(K)/\mu(K))| < c_1\,\mathrm{vol}(K) + c_2~~$$

$$|2\sigma(K) - \mathrm{Re}\,(\lambda(K)/\mu(K))| < c\,\mathrm{vol}(K)\ \mathrm{inj}(K)^{-3}$$

which was turned into a theorem in a separate publication [4]!

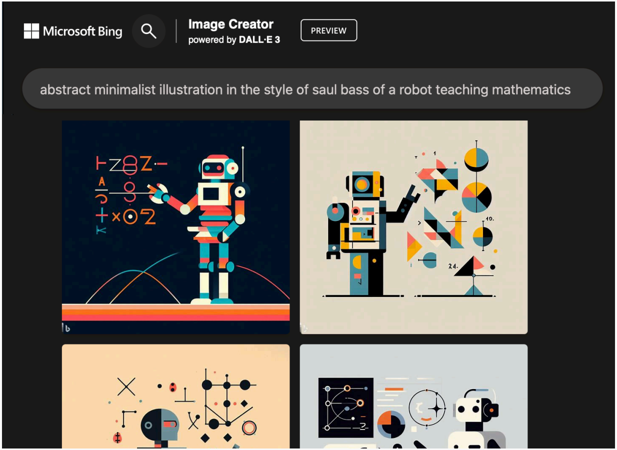
Emergence

Emergence

Over the last few years we have increasingly seen systems transcending big data with capabilities that verge on emergence.

Emergence

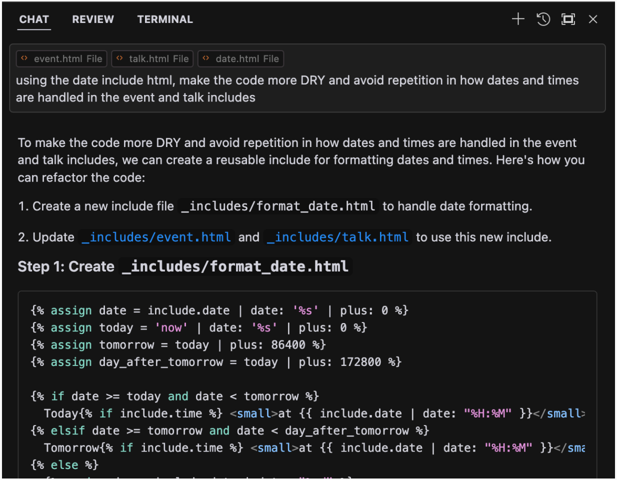
Over the last few years we have increasingly seen systems transcending big data with capabilities that verge on emergence.



abstract minimalist illustration in the style of saul bass of a robot teaching mathematics

Emergence

Over the last few years we have increasingly seen systems transcending big data with capabilities that verge on emergence.



using the date include html, make the code more DRY and avoid repetition in how dates and times are handled in the event and talk includes

To make the code more DRY and avoid repetition in how dates and times are handled in the event and talk includes, we can create a reusable include for formatting dates and times. Here's how you can refactor the code:

1. Create a new include file `_includes/format_date.html` to handle date formatting.
2. Update `_includes/event.html` and `_includes/talk.html` to use this new include.

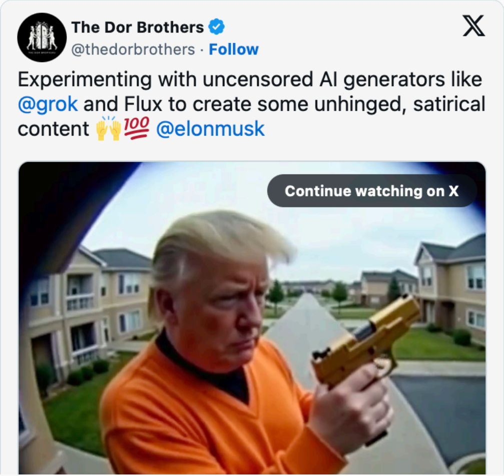
Step 1: Create `_includes/format_date.html`

```
{% assign date = include.date | date: '%s' | plus: 0 %}
{% assign today = 'now' | date: '%s' | plus: 0 %}
{% assign tomorrow = today | plus: 86400 %}
{% assign day_after_tomorrow = today | plus: 172800 %}

{% if date >= today and date < tomorrow %}
  Today{% if include.time %} <small>at {{ include.date | date: "%H:%M" }}</small
{% elsif date >= tomorrow and date < day_after_tomorrow %}
  Tomorrow{% if include.time %} <small>at {{ include.date | date: "%H:%M" }}</sma
{% else %}
```

Emergence

Over the last few years we have increasingly seen systems transcending big data with capabilities that verge on emergence.



The Dor Brothers
@thedorbrothers · Follow

Experimenting with uncensored AI generators like @grok and Flux to create some unhinged, satirical content 🙌💯 @elonmusk

Emergence

Why has theoretical math not been affected by this (yet)?

Emergence

Why has theoretical math not been affected by this (yet)?

Large Language Models (LLMs) have been called stochastic parrots that do not transcend the quality of their input.

Emergence

Why has theoretical math not been affected by this (yet)?

Large Language Models (LLMs) have been called stochastic parrots that do not transcend the quality of their input. Unlike (some) humans, they do not internally reason before answering.

Emergence

Why has theoretical math not been affected by this (yet)?

Large Language Models (LLMs) have been called stochastic parrots that do not transcend the quality of their input. Unlike (some) humans, they do not internally reason before answering.

Computation has seemingly gone from *precise output when given precise input* to accepting *fuzzy input at the cost of fuzzy output*.

Emergence

Why has theoretical math not been affected by this (yet)?

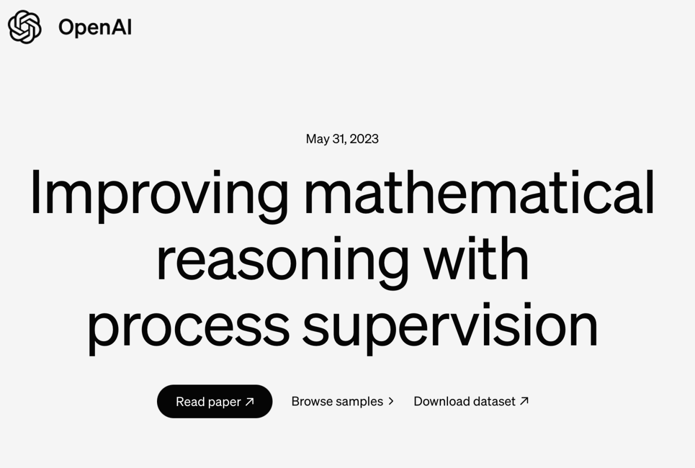
Large Language Models (LLMs) have been called stochastic parrots that do not transcend the quality of their input. Unlike (some) humans, they do not internally reason before answering.

Computation has seemingly gone from *precise output when given precise input* to accepting *fuzzy input at the cost of fuzzy output*.

But this story is not entirely true ...

Emergence

Math has become a proving ground for AI companies claiming to have figured out reasoning while avoiding ‘hallucinations’.



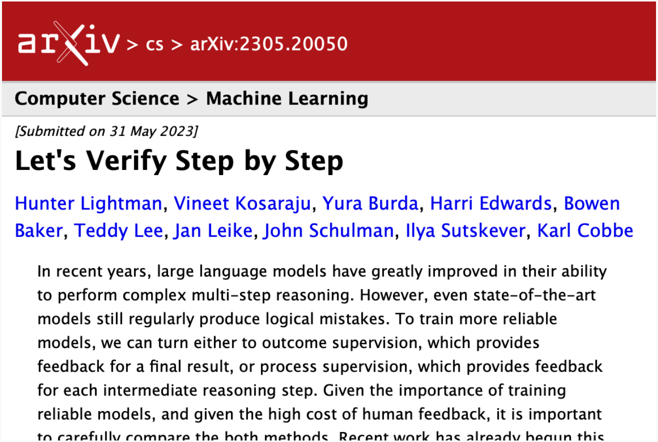
OpenAI

May 31, 2023

Improving mathematical reasoning with process supervision

Emergence

Math has become a proving ground for AI companies claiming to have figured out reasoning while avoiding ‘hallucinations’.



arXiv > cs > arXiv:2305.20050

Computer Science > Machine Learning

[Submitted on 31 May 2023]

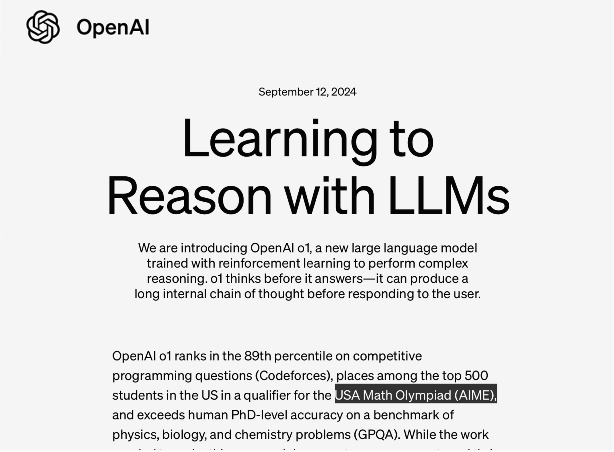
Let's Verify Step by Step

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, Karl Cobbe

In recent years, large language models have greatly improved in their ability to perform complex multi-step reasoning. However, even state-of-the-art models still regularly produce logical mistakes. To train more reliable models, we can turn either to outcome supervision, which provides feedback for a final result, or process supervision, which provides feedback for each intermediate reasoning step. Given the importance of training reliable models, and given the high cost of human feedback, it is important to carefully compare the both methods. Recent work has already begun this

Emergence

Math has become a proving ground for AI companies claiming to have figured out reasoning while avoiding ‘hallucinations’.



OpenAI

September 12, 2024

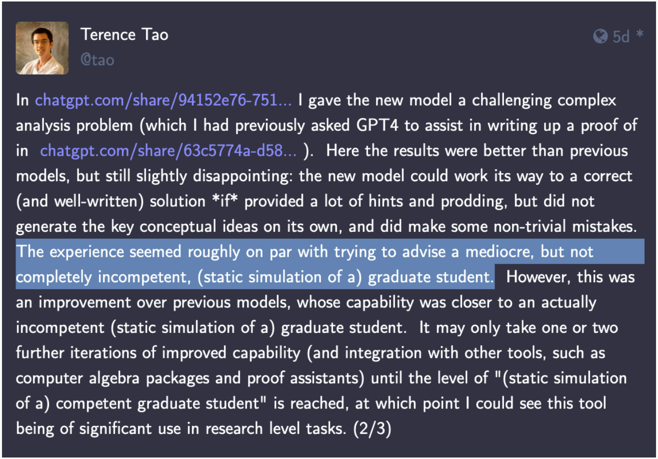
Learning to Reason with LLMs

We are introducing OpenAI o1, a new large language model trained with reinforcement learning to perform complex reasoning. o1 thinks before it answers—it can produce a long internal chain of thought before responding to the user.

OpenAI o1 ranks in the 89th percentile on competitive programming questions (Codeforces), places among the top 500 students in the US in a qualifier for the USA Math Olympiad (AIME), and exceeds human PhD-level accuracy on a benchmark of physics, biology, and chemistry problems (GPQA). While the work

Emergence

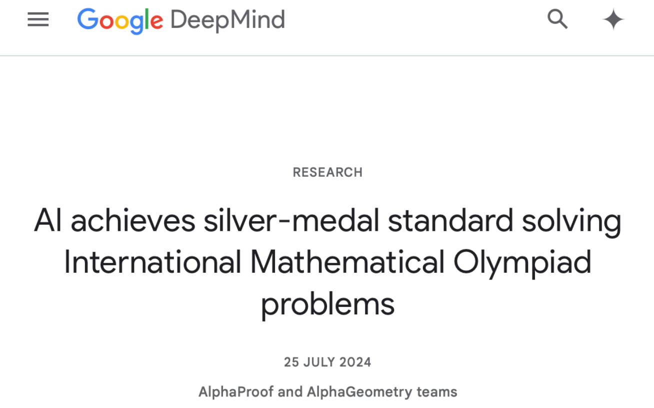
Math has become a proving ground for AI companies claiming to have figured out reasoning while avoiding ‘hallucinations’.



In chatgpt.com/share/94152e76-751… I gave the new model a challenging complex analysis problem (which I had previously asked GPT4 to assist in writing up a proof of in chatgpt.com/share/63c5774a-d58…). Here the results were better than previous models, but still slightly disappointing: the new model could work its way to a correct (and well-written) solution *if* provided a lot of hints and prodding, but did not generate the key conceptual ideas on its own, and did make some non-trivial mistakes. The experience seemed roughly on par with trying to advise a mediocre, but not completely incompetent, (static simulation of a) graduate student. However, this was an improvement over previous models, whose capability was closer to an actually incompetent (static simulation of a) graduate student. It may only take one or two further iterations of improved capability (and integration with other tools, such as computer algebra packages and proof assistants) until the level of "(static simulation of a) competent graduate student" is reached, at which point I could see this tool being of significant use in research level tasks. (2/3)

Emergence

Math has become a proving ground for AI companies claiming to have figured out reasoning while avoiding ‘hallucinations’.



Google DeepMind

RESEARCH

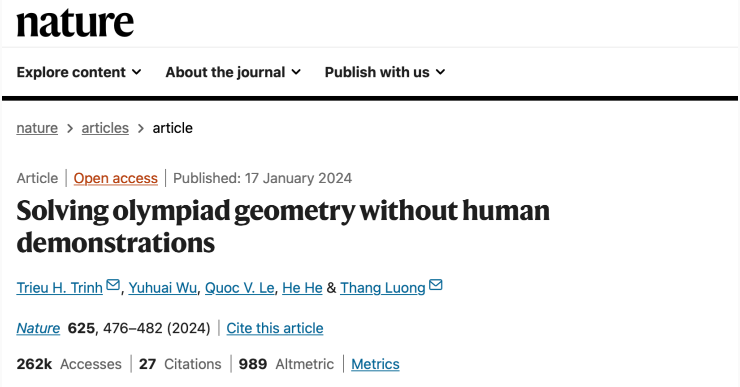
AI achieves silver-medal standard solving International Mathematical Olympiad problems

25 JULY 2024

AlphaProof and AlphaGeometry teams

Emergence

Math has become a proving ground for AI companies claiming to have figured out reasoning while avoiding ‘hallucinations’.



nature

nature > articles > article

Article | Open access | Published: 17 January 2024

Solving olympiad geometry without human demonstrations

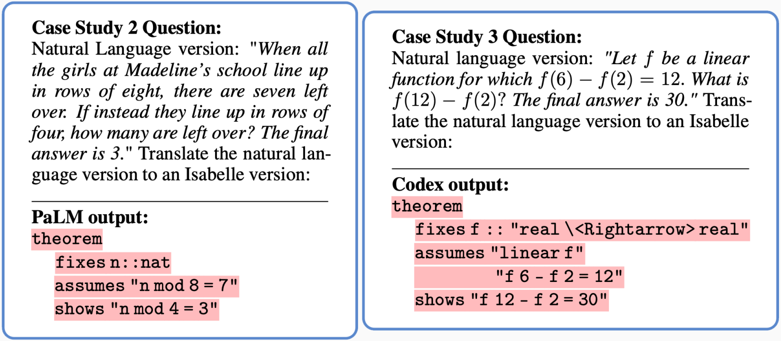
Trieu H. Trinh, Yuhuai Wu, Quoc V. Le, He He & Thang Luong

Nature **625**, 476–482 (2024) | Cite this article

262k Accesses | **27** Citations | **989** Altmetric | Metrics

Emergence

Many approaches work directly in formal proof languages…



Case Study 2 Question:
Natural Language version: "*When all the girls at Madeline's school line up in rows of eight, there are seven left over. If instead they line up in rows of four, how many are left over? The final answer is 3.*" Translate the natural language version to an Isabelle version:

PaLM output:

```
theorem
  fixes n::nat
  assumes "n mod 8 = 7"
  shows "n mod 4 = 3"
```

Case Study 3 Question:
Natural language version: *"Let f be a linear function for which $f(6) - f(2) = 12$. What is $f(12) - f(2)$? The final answer is 30."* Translate the natural language version to an Isabelle version:

Codex output:

```
theorem
  fixes f :: "real \<Rightarrow> real"
  assumes "linear f"
          "f 6 - f 2 = 12"
  shows "f 12 - f 2 = 30"
```

Figure 3: Autoformalizations from natural language to Isabelle code. **Left:** Case study 2 – perfect formalization by PaLM. **Right:** Case study 3 – incorrect formalization by Codex.

Figure taken from [18]

Pointers for your own research

Pointers for your own research

- Implementing a machine learning approach is often **hard** for entirely different reasons than mathematics is hard.

Pointers for your own research

- Implementing a machine learning approach is often **hard** for entirely different reasons than mathematics is hard.
- Know in advance if you care about the application or method.

Pointers for your own research

- Implementing a machine learning approach is often **hard** for entirely different reasons than mathematics is hard.
- Know in advance if you care about the application or method.
- Be realistic in you expectations of what you can achieve.

Pointers for your own research

- Implementing a machine learning approach is often **hard** for entirely different reasons than mathematics is hard.
- Know in advance if you care about the application or method.
- Be realistic in you expectations of what you can achieve.
- If you end up publicizing your method, be aware of existing approaches (in particular beyond Machine Learning):

Pointers for your own research

- Implementing a machine learning approach is often **hard** for entirely different reasons than mathematics is hard.
- Know in advance if you care about the application or method.
- Be realistic in you expectations of what you can achieve.
- If you end up publicizing your method, be aware of existing approaches (in particular beyond Machine Learning):
  - Discrete black-box optimization has been studied since the 60s with effective approaches like Simmulated Annealing [10]. They have been extensively used to find objects like Ramsey colorings.

Pointers for your own research

- Implementing a machine learning approach is often **hard** for entirely different reasons than mathematics is hard.
- Know in advance if you care about the application or method.
- Be realistic in you expectations of what you can achieve.
- If you end up publicizing your method, be aware of existing approaches (in particular beyond Machine Learning):
    - Discrete black-box optimization has been studied since the 60s with effective approaches like Simmulated Annealing [10]. They have been extensively used to find objects like Ramsey colorings.
    - Graffiti [6] or AutoGraphiX [8] have been formulating and refuting conjectures in extremal graph theory since the 80s.

Thank you!

References

[1] M. Aouchiche and P. Hansen.
A survey of automated conjectures in spectral graph theory.
Linear algebra and its applications, 432(9):2293–2322, 2010.

[2] M. Aouchiche and P. Hansen.
Proximity, remoteness and distance eigenvalues of a graph.
Discrete Applied Mathematics, 213:17–25, 2016.

[3] R. A. Brualdi and L. Cao.
Pattern-avoiding (0, 1)-matrices.
arXiv preprint arXiv:2005.00379, 2020.

[4] A. Davies, A. Juhász, M. Lackenby, and N. Tomasev.
The signature and cusp geometry of hyperbolic knots.
arXiv preprint arXiv:2111.15323, 2021.

References

[5] A. Davies, P. Veličković, L. Buesing, S. Blackwell, D. Zheng, N. Tomašev, R. Tanburn, P. Battaglia, C. Blundell, A. Juhász, et al.
Advancing mathematics by guiding human intuition with ai.
Nature, 600(7887):70–74, 2021.

[6] S. Fajtlowicz.
On conjectures of graffiti.
In *Annals of Discrete Mathematics*, volume 38, pages 113–118. Elsevier, 1988.

[7] A. Fawzi, M. Balog, A. Huang, T. Hubert, B. Romera-Paredes, M. Barekatain, A. Novikov, F. J. R Ruiz, J. Schrittwieser, G. Swirszcz, et al.
Discovering faster matrix multiplication algorithms with reinforcement learning.
Nature, 610(7930):47–53, 2022.

References

[8] P. Hansen and G. Caporossi.
Autographix: An automated system for finding conjectures in graph theory.
Electronic Notes in Discrete Mathematics, 5:158–161, 2000.

[9] C. Hertrich, A. Basu, M. Di Summa, and M. Skutella.
Towards lower bounds on the depth of relu neural networks.
Advances in Neural Information Processing Systems, 34:3336–3348, 2021.

[10] S. Kirkpatrick, C. D. Gelatt Jr, and M. P. Vecchi.
Optimization by simulated annealing.
science, 220(4598):671–680, 1983.

References

[11] A. Mehrabian, A. Anand, H. Kim, N. Sonnerat, M. Balog, G. Comanici, T. Berariu, A. Lee, A. Ruoss, A. Bulanova, et al.
Finding increasingly large extremal graphs with alphazero and tabu search.
arXiv preprint arXiv:2311.03583, 2023.

[12] K. Mundinger, S. Pokutta, C. Spiegel, and M. Zimmer.
Extending the continuum of six-colorings.
Geombinatorics Quarterly, XXXIV, 2024.

[13] S. Polu and I. Sutskever.
Generative language modeling for automated theorem proving.
arXiv preprint arXiv:2009.03393, 2020.

References

[14] B. Romera-Paredes, M. Barekatain, A. Novikov, M. Balog, M. P. Kumar, E. Dupont, F. J. Ruiz, J. S. Ellenberg, P. Wang, O. Fawzi, et al.
Mathematical discoveries from program search with large language models.
Nature, 625(7995):468–475, 2024.

[15] Z. Shen, H. Yang, and S. Zhang.
Optimal approximation rate of relu networks in terms of width and depth.
Journal de Mathématiques Pures et Appliquées, 157:101–135, 2022.

[16] T. H. Trinh, Y. Wu, Q. V. Le, H. He, and T. Luong.
Solving olympiad geometry without human demonstrations.
Nature, 625(7995):476–482, 2024.

References

[17] A. Z. Wagner.
Constructions in combinatorics via neural networks.
arXiv preprint arXiv:2104.14516, 2021.

[18] Y. Wu, A. Q. Jiang, W. Li, M. Rabe, C. Staats, M. Jamnik, and C. Szegedy.
Autoformalization with large language models.
Advances in Neural Information Processing Systems, 35:32353–32368, 2022.