# Some challenges of AI to Mathematical Research

April 2023

## 1 Introduction

The purpose of this article is to explore and report on the state-of-the-art in machine learning for *mathematical research*. There are well-written surveys that cover other aspects of deep learning in mathematics, such as Lu et al. [17]. The article, in spirit, serves as an extension to Williamson's survey [37], but with an addition in the progress of generative AI and the theorem-proving community.

Mathematics, as a societal endeavor, encompasses teaching, learning, research, publishing, and outreach. There are complex dynamics between different stakeholders. Corporate interests often diverge from academic research Academic research, on the other hand, is bounded by exhaustive publication protocols. The social nature of Mathematics often transcends beyond private thoughts or perceptions; and its utility can raise crucial ethical questions, [6]. However, due to constraints, we curtail our discussion here. We refer to the text [13] for an introduction to these ideas.

The structure of the article follows that of a research mathematician, as described by Atyiah, [10]

> In mathematics, ideas and concepts come first, then come questions and problems. At this stage the search for solutions begins, one looks for a method or strategy...Before long you may realize, perhaps by finding counterexamples, that the problem was incorrectly formulated... Without proof the program remains incomplete, but without the imaginative input it never gets started.

A survey of an online collaborative project [19] reflects this distribution of mathematical research, see Fig. 1,

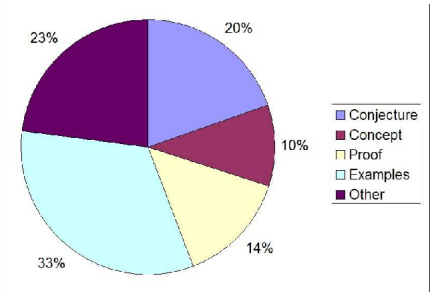



Figure 1: Distribution of comment contents

As observed by others, solving the challenge of AI-enabled mathematics not only helps mathematicians in their research but it also serves as a litmus test for AGI itself: the ability to reason which is at the core of mathematical enquiry is a central unsolved problem in artificial intelligence.
In (2), we discuss current AI progress towards the first stages of conjecturing. In (3) we discuss progress towards the formalization aspects of proofs. Lastly, in (4), we propose ways by which we integrate these two.

### 1.1 Language models for mathematics

Recent advances and successes of large language models suggests exciting avenue. With transfer learning, one can leverage the pre-trained model's knowledge to achieve better performance on the target task, even with limited labeled data. This led to a range of pre-trained language models on mathematical data and fine tuned on mathematical tasks. Language models satisfy the following two important properties:

- scaling laws. The ability of model scales with both the number of parameters and the training corpus. The availability of good quality training corpus compared to natural language is scarce. In terms of training corpus, there are generally two types

1. Curated datasets, such as Hendryck et al's [12], GitHuB, or arXiv datasets
2. Synthetic datasets. Recent works, [38], suggests improvements from training on synthetic datasets.

- in-context learning.

The extent to which mathematicians can leverage language models will be the focal point of this survey paper. Tao gives a succinct summary of the current state of the art in using LLM generated materials [31].

> Both humans and AI need to develop skills to analyze this new type of text. The stylistic signals that I traditionally rely on to "smell out" a hopelessly incorrect math argument are of little use with LLM-generated mathematics. Only line-by-line reading can discern if there is any substance. Strangely, even nonsensical LLM-generated math often references relevant concepts. With effort, human experts can modify ideas that do not work as presented into a correct and original argument.

## 2 Conjectures, examples and tests

When trying to digitize mathematics, it can be perceived that mathematics and the sciences differ in their ontological (what are the objects being studied?), generative (how are examples created?) and epistemological(how do we justify?) aspects. This is seen in the table [1]

| Mathematics | Sciences |
|---|---|
| Symbolic methods | Neural methods |
| Automated reasoning | Machine learning |
| Formal methods | Data science |

This point of view is explored further in (3).

In this section, we perceive mathematics more as physical sciences and discuss how machine learning aids in generating *meaningful* conjectures. A crude form of this is exemplified in the enlightening book, *Proofs and Refutations* of Lakotos [16], which has thus form the grounding of modeling mathematical discourse, [18] and mathematics education, [15]. Lakatos imagines an imaginary classroom, where the teacher makes a conjecture of the Euler-Poincaré formula for polyhedras: given a polyhedron where $V$ is number of vertices, $E$ is number of edges and $F$ is number of faces, then

$$V - E + F = 2 \tag{1}$$

The teacher provides a proof where students attempt to refute it by providing counterexamples. The dialogue continues back and forth, and the process can be summarized as

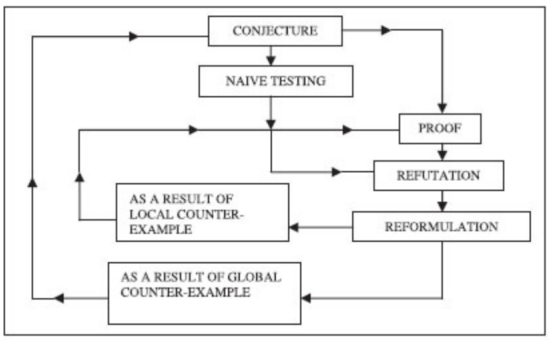



Figure 2: Lakatos method as explained in [8]

The key elements which are cycled in this process will form the basis of this section.

1. Generating a conjecture, can we make computations?
2. Testing and refutation

In a somewhat similar spirit, Harris' blog post, [11], made an intriguing thought exercise of whether one can recreate the Poincaré conjecture in Thurston's paper [32]
Lastly, we raise the following question:

> can one create synthetic datasets of mathematical discourse (for research) and if so, how can we evaluate a model trained on such data?

## 2.1 Generating conjectures and new concepts

One of the more prominent approaches to generate hypothesis, follow the following pipeline [7]

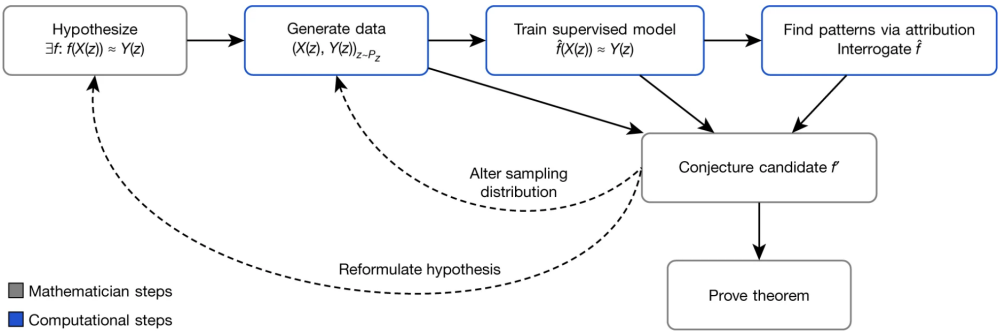



Figure 3: Pipeline for hypothesis generation in [7].

To illustrate this, consider the example of Euler poincaré conjecture, 1. One begins at the left most *blue* box. The *grey* box are required by human intervention. One conjectures the relation between various features of polyhedra $z$ sample from a distribution. Let

$$X(z) = (V(z), E(z), \text{Vol}(z), \text{Sur}(z))$$

the number of vertices, number of edges, volume and surface area of $z$.

$$Y(z) = (F(z))$$

be the number faces of $z$. Our goal is to find a function $\hat{f}(X(z)) = Y(z)$. Using classical supervised learning, we can obtain

$$\hat{f}((X(z)) = X(z) \cdot (1, -1, 0, 0) + 2$$

However, when data is high dimensional and underlying relation is non-linear, $\hat{f}$ is only there for *intution*. The outcome of this training process is thus accompanied with *attribution techniques* , such as *gradient based techniques*, [29]. Another key aspect of mathematics is creating definitions, [33]. [1] Examples of recent creations include the notion of *quasicategories* [24] and *perfectoid spaces* [26]. Could AI create new and meaningful objects? For example, the Ramanujan machine, [21].

### 2.1.1 Machine learning for experimentation

There are many unexpected connections in mathematics. Most of these come from *experimentation* and *computation*. In number theory, one has the *j-invariant* function, which is a *modular function* on the upper half plane $\mathbb{H} := \{\tau \in \mathbb{C} : \text{im}\tau > 0\}$.

$$j(\tau) := \frac{1}{q} + 196884q + 2149370q^2 + \cdots \qquad \text{where } q = e^{2\pi i \tau}$$

This is an object of close connection to *elliptic curves*, [27]. John-McKay found that these coefficients have close relations with dimensions of irreducible representations of Monster groups (which led to subsequent work vertex operator algebras.) This involved many linear algebra computations.

There has been recent research in the direction of progressing experimentation, particularly the work of Charton, [4] which trains transformers to perform numerical computation, and [5] which trains the model to learn mathematical properties of differential systems and finding properties of algebraic objects [2].

## 2.2 Conjecture verification

Once a conjecture is made, the first step would often be a search on whether similar results have been proven. Advances in *semantic search* can help propel this field. For instance, a typical query would be of the form: "Is it true that xxx is satisfied for yyy". Such open-ended questions are hard. It is also unlikely that the same phrasing or even words were used in the hypothetical reference. Progress towards autoformalization, 3.5,

[1] This is in contrast to daily life. In Fodor et al's paper "against definition", the paper discusses how in the non-technical language one does not actually work with definitions.

could help with this. Below we give two instances of which methods in machine learning can *aid* in disproving conjectures.
In the field of solving PDE, neural networks has already been used in closely related context of physics, [23]. For instance, The physics of an equation is explained by a PDE

$$P(u(x,t)) = 0$$

where $u$ is a function of location $x$ and time $t$. The goal is then to apply supervised learning with and solve the function $u$. This has inspired many subsequent works. A recent example,[35], [2] Then we defines loss via ... ? Thisi s the basis of Physics-informed neural networks (PINNs). It has only be successful in identifying the Reynolds numbers from given flow an the Navier-Stokes equation. Otherworks, such as Wagner's, [34], have used reinforcement learning to guide intuition in combinatorics. Importantly, it may not generate the counter-example, but gives sufficient intuition for counter-examples, we refer the discussion to [37, 6].

# 3 Deductive reasoning

## 3.1 The Origin of Formal Mathematics

Historically, the idea of inferring mathematical truths through a system of logical deductions goes back to Aristotle (c.f. Prior Analytics). In the 17th century, the polymath Leibniz (1647-1716) dreamed of a universal mathematical language, *Calculus Ratiocinator*, or the *Calculus of thoughts*, [25], where logical and mathematical truths could be encoded and systematically deciphered. George Boole (1815-1864) took the discipline a significant step forward by introducing an algebraic treatment to Aristotle's syllogisms, leading to the groundwork of Boolean algebras. Soon, Gottlob Frege initiated the movement of *logicism* with his seminal work "Begriffsschrift" (1879). The work of Leibniz in the 17th century and Frege in the 19th century expanded the logical system and thereby allowed vaster domains of mathematics to be accessible to purely logical derivations.

The formalist approach led to the search for secure foundations for mathematics and ZFC set theory and type theory emerged as the two main foundations of mathematics. In the foundationalist approach to mathematics, every mathematical object is constructed from the most basic objects such as the type of natural numbers subject. The manipulations of objects and elements in them are then subject to the rules of logic. For several important reasons, type theory, has been favored in formalizing mathematics in computers.

We remark that, for a long while, the foundationalist approach seemed to be at odds with the structuralist view of mathematics (e.g. the mathematics of Riemann, Dedekind, Hilbert, Bourbaki, and Grothendieck). This was mainly due to the ergonomical limitations of ZFC set theory in allowing a faithful encoding of mathematical structures. Often the encoding of mathematical structures (e.g. manifolds) are completely unrecognizable from their intuitive and informal definitions. Set theory although a complete foundation for mathematics, distorts the structures in the process of formalizing them: think of mathematical structures as specifications in programming language with high-level abstractions and the set theoretic encoding as binary code after compilation. We can learn little from the binary code about what the programs do.

Dependent Type Theory and Homotopy Type Theory offer a vastly superior encoding of mathematical structures. They are implemented in various Interactive Proof Assistants (ITP) such as Lean, Coq, Agda, Hol Light, Isabelle, etc where verification is type checking. Lean, in particular, has gained huge attention from the formalization of Liquid Tensor Experiment , [3] which proved a difficult theorem of the cutting-edge research in algebraic geometry.

## 3.2 Automated Reasoning

Automatic theorem provers (ATPs) have grown in past decades, and is used in conjuction with interactive proof assistants ITPs, 3.3, such as [9]. Historically, emphasis have been placed on SAT or Satisfyability modulo theories (SMT) problems, mainly focusing imporving the efficiency and performance.[3] One fundamental task is *premise selection*, [22]. SAT solvers are employed in Lean and Isabeller to increase user's productivity: for instance automating long chains of reasoning steps to prove formulas (c.f. tactic `tauto` in Lean), as well as reasoning with and about equations and inequalities, and various algebraic decision procedures. In Isablle external first-order provers can be invoked through sledgehammer.

## 3.3 Proof assistant

Interactive Theorem Provers (ITPs) are reasoning engines for formalization of mathematical proofs. Various ITPs have been developed over the past decades, notably Mizar (1973), Isabelle (1994), Coq (1997), and more

[2] For instance, the viscous Burger's equation is $\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = v\frac{\partial^2 u}{\partial x^2}$
[3] ATPS are often based on first-order logic, which limits their expressibility.

popular recently [4], Lean (2013).

These are useful for:

- mathematical knowledge management.

As of now ITP does not involve AI. A key part of the use of language model can be in *user experience*

- Desirably, formal proofs should be : flexible, powerful and expressive.
- search of terms.

Over the last few years, we have seen breakthroughs in large-scale deployment of Interactive theorem provers (ITPs) in the cutting-edge mathematical research such as the Liquid Tensor Experiment [3]. [5]

A fanciful dream is that formalization and proof verification will combine well with LLMs and we will end up with computers being better than humans at mathematical research. For many technical reasons around formal verification this is currently science fiction and we conjecture it will remain so in the foreseeable future.

Despite the fact that LLMs are not good at reasoning (at least at the undergraduate university level) LLMs are still very useful when formalizing mathematics. While LLMs are not good at reasoning, they are great tools for pattern matching and this feature can be well combined with the architecture of the proof assistants.

We already have Segredo as a tool to utilize ChatGPT to suggest next proof steps inside a proof tactic bloc in Lean. Another preliminary tool developed by Adam Topaz used ChatGPT and the word embedding. Here's how it works: say we are interested in invoking a formal lemma in our proof but you don't know the name of it in the vast Lean's mathematical library (Mathlib). We state the lemma in the natural language (informally), and the tool uses ChatGPT to hallucinate as much as possible and spit out formal statements which are possible matches above certain matching threshold.

Mathematics is a great playground for abstract reasoning. Some have suggested that "solving mathematics" is a necessary prerequisite for AGI. However, this is hard! Consider the following analogy: Computers became better than humans at chess in the 1990s. Computers became vastly better than humans at all board games in the last few years.

Maths is a game with well-defined rules. The key difference is that although the goals in the board games are well-defined that is not the case in math.

The main problem with training neural network models for formalized proofs is the lack of data. However, certain techniques have been developed to compensate to overcome this difficulty. Given fixed compute training budget, it has been suggested that the expert iteration outperforms proof search only (e.g. Lean GPT-f which solved IMO problems with 36 layers and 774 million trainable parameters).

### 3.4 Shortcomings of formal proofs

In the practice of mathematics lots of time spent on examples, concepts and conjectures. These are essential mathematical activities which are not faithfully reflected in the final formal libraries mathematics.

We need AI tools to help us with suggesting examples, forming and testing conjectures, and arriving at concepts. This will be significantly more challenging than the task of Auto-formalization or proof automation as it will be harder to come up with good benchmark for examples/concepts/conjectures than for the proofs.

### 3.5 Autoformalization

Autoformalization is the task of turning informal descriptions to formal mathematics. Examples of formalization include Kepler conjecture, Four-Color theorem and Feit-Thompson theorem, giving certainty to the correctedness.

The main remaining challenge of autoformalization is the misalignment of the informal mathematical definitions and the formal library code. The other challenges

is extremely challenging in the sense that the model needs to (1) bridge the logical gaps left in pen-and-paper proofs, (2) assume the implicit contexts and assumptions, and (3) align informal definitions/concepts to formal ones.

Autoformalization is seen as essential, [30, 3] in training language model. Compared to large body of corpus on the web, the Archive of Formal Proofs consists of less than 0.5% of the training data than large language model like Codex. Recent progress uses large language models, Wu et al. [38], which contrast to older series of work [20].

[4] Currently, there are approximately 25 people managing its extensively library, mathlib, which contains more than $10^6$ lines of code.

[5] LTE is the first test that proof assistants can organize mathematical proofs at scale. The complexity of LTE is way more than anything we had before since it relies on formalization of many results from different fields of mathematics.

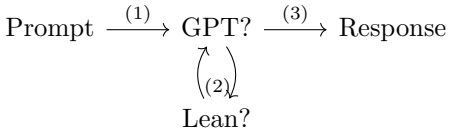
### 3.5.1 Autoformalizers and multimodal learning

Proof by analogies: There are lots of words like "analogously" and "similarly in informal math that mathematicians broadly understand. But what if they don't understand the precise meaning of those in the context of particular proofs when looking back in the future? Here is a challenge: How do we formalize proofs by similarity or even more difficult analogical proofs?

We propose a useful application of multi-modal learning in creating a Lean plugin whereby we have an auto-formalizer for proofs-from-pictures. Often mathematicians employ diagrams and pictures to convey key points in their proofs. Can we build a tool which turns a picture into a formal code?

# 4 Future directions: the scientific coauthor

As suggested in a similar form by a professor of Mathematics, Alex Kontrovich, and already existing in many fields of research, [28] , a natural holy grail is to have a scientific coauthor. A typical pipeline should look like:




In (1), the reader suggests an idea. In (2) the model runs inferences and checks against itself using interactive theorem provers. In (3) it makes it final inference. Lastly, the process is cycled.
Combining all of (3) , (2) into one seamless pipeline is ambitious and is far from current capabilities. Works in this direction include, *Draft, Sketch and Proof*, of Jiang et al, [14], and LeanDojo. [6] To name a few further desirable traits: proof explaining/outlining, proof repair, library design.

Lastly, whether in use of autoformalization, 3.5, or response generation, as 2, it is desirable that the output to be sound and reliable. LLM outputs can have unwanted repetition, [39], lack of generalization, lack of robustness [36], and even non-human interpretable, as can be seen in [38]. Some future avenues are suggested in [17, 7.2].

[6] Unfortunately, this still has many drawbacks.

# References


[1] Jeremy Avigad. Is mathematics obsolete?, 2023.

[2] Jiakang Bao, Yang-Hui He, Edward Hirst, Johannes Hofscheier, Alexander Kasprzyk, and Suvajit Majumder. Hilbert series, machine learning, and applications to physics. *Physics Letters B*, 827:136966, 2022.

[3] Reid Barton, Johan Commelin, Patrick Massot, Scott Morrison, Adam Topasz, and Peter Scholze. Liquid tensor experiment, 2021.

[4] François Charton. Linear algebra with transformers, 2022.

[5] François Charton, Amaury Hayat, and Guillaume Lample. Learning advanced mathematical computations from examples, 2021.

[6] Maurice Chiodo and Toby Clifton. The importance of ethics in mathematics. *European Mathematical Society Magazine*, (114):34–37, 2019.

[7] Alex Davies, Petar Veličković, Lars Buesing, Sam Blackwell, Daniel Zheng, Nenad Tomašev, Richard Tanburn, Peter Battaglia, Charles Blundell, András Juhász, et al. Advancing mathematics by guiding human intuition with ai. *Nature*, 600(7887):70–74, 2021.

[8] Philip J Davis, Reuben Hersh, and Elena Anne Marchisotto. *The mathematical experience, study edition.* Springer, 2012.

[9] Burak Ekici, Guy Katz, Chantal Keller, Alain Mebsout, Andrew J. Reynolds, and Cesare Tinelli. Extending SMTCoq, a certified checker for SMT (extended abstract). *Electronic Proceedings in Theoretical Computer Science*, 210:21–29, jun 2016.

[10] Timothy Gowers, June Barrow-Green, and Imre Leader. Viii.6 advice to a young mathematician. 2010.

[11] Michael Harris. What is "human-level mathematical reasoning"?, part 1: which humans?, Nov 2021.

[12] Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset, 2021.

[13] Reuben Hersh. What is mathematics, really? *Mitteilungen der Deutschen Mathematiker-Vereinigung*, 6(2):13–14, 1998.

[14] Albert Qiaochu Jiang, Sean Welleck, Jin Peng Zhou, Wenda Li, Jiacheng Liu, Mateja Jamnik, Timothée Lacroix, Yuhuai Wu, and Guillaume Lample. Draft, sketch, and prove: Guiding formal theorem provers with informal proofs. *ArXiv*, abs/2210.12283, 2022.

[15] Fatih Karakuş and Mesut Bütün. Examining the method of proofs and refutations in pre-service teachers education. *Bolema: Boletim de Educação Matemática*, 27:215–232, 2013.

[16] Imre Lakatos. *Proofs and refutations: The logic of mathematical discovery.* Cambridge university press, 2015.

[17] Pan Lu, Liang Qiu, Wenhao Yu, Sean Welleck, and Kai-Wei Chang. A survey of deep learning for mathematical reasoning, 2023.

[18] Alison Pease, John Lawrence, Katarzyna Budzynska, Joseph Corneli, and Chris Reed. Lakatos-style collaborative mathematics through dialectical, structured and abstract argumentation. *Artificial Intelligence*, 246:181–219, 2017.

[19] Alison Pease and Ursula Martin. Seventy four minutes of mathematics: An analysis of the third mini-polymath project. 2012.

[20] Stanislas Polu, Jesse Michael Han, Kunhao Zheng, Mantas Baksys, Igor Babuschkin, and Ilya Sutskever. Formal mathematics statement curriculum learning, 2022.

[21] Gal Raayoni, Shahar Gottlieb, Yahel Manor, George Pisha, Yoav Harris, Uri Mendlovic, Doron Haviv, Yaron Hadad, and Ido Kaminer. Generating conjectures on fundamental constants with the ramanujan machine. *Nature*, 590(7844):67–73, feb 2021.

[22] Markus N. Rabe, Dennis Lee, Kshitij Bansal, and Christian Szegedy. Mathematical reasoning via self-supervised skip-tree training, 2020.

[23] Maziar Raissi, Paris Perdikaris, and George E Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational physics*, 378:686–707, 2019.

[24] Charles Rezk. Introduction to quasicategories. *Lecture Notes for course at University of Illinois at Urbana-Champaign*, 2022.

[25] Daniel M. Rice. Calculus of thought: Neuromorphic logistic regression in cognitive machines. 2013.

[26] Peter Scholze. Perfectoid spaces. *Publications mathématiques de l'IHÉS*, 116(1):245–313, 2012.

[27] Joseph H Silverman. *The arithmetic of elliptic curves*, volume 106. Springer, 2009.

[28] Chris Stokel-Walker. Chatgpt listed as author on research papers: many scientists disapprove. *Nature*, 613:620–621, 2023.

[29] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International conference on machine learning*, pages 3319–3328. PMLR, 2017.

[30] Christian Szegedy. A promising path towards autoformalization and general artificial intelligence. In *International Conference on Intelligent Computer Mathematics*, 2020.

[31] Terence Tao. Embracing change and resetting expectations, 2023.

[32] William P Thurston. Three dimensional manifolds, kleinian groups and hyperbolic geometry. 1982.

[33] Shlomo Vinner. The role of definitions in the teaching and learning of mathematics. 2002.

[34] Adam Zsolt Wagner. Constructions in combinatorics via neural networks, 2021.

[35] Yongji Wang, Ching-Yao Lai, Javier Gómez-Serrano, and Tristan Buckmaster. Asymptotic self-similar blow-up profile for three-dimensional axisymmetric euler equations using neural networks, 2023.

[36] Sean Welleck, Peter West, Jize Cao, and Yejin Choi. Symbolic brittleness in sequence models: on systematic generalization in symbolic mathematics, 2022.

[37] Geordie Williamson. Is deep learning a useful tool for the pure mathematician?, 2023.

[38] Yuhuai Wu, Albert Q. Jiang, Wenda Li, Markus N. Rabe, Charles Staats, Mateja Jamnik, and Christian Szegedy. Autoformalization with large language models, 2022.

[39] Jin Xu, Xiaojiang Liu, Jianhao Yan, Deng Cai, Huayang Li, and Jian Li. Learning to break the loop: Analyzing and mitigating repetitions for neural text generation. *ArXiv*, abs/2206.02369, 2022.