Two approaches to developing human-like artificial mathematical intelligence

Markus Pantsar (RWTH Aachen University)

Pre-print

Abstract

Mathematics has been an important topic in artificial intelligence (AI) research already from the very beginning. In recent discussions, however, mathematics is not seen as part of the success stories in AI. While AI tools are used in mathematical practice, they are limited to rule-based systems with limited applications. In this paper, I explore two emerging machine-learning based approaches to developing an AI system that could prove mathematical theorems that are interesting to human mathematicians. In the top-down approach, the AI is trained with mathematical texts, as is done in the training of large language models. In the bottom-up approach, the AI is developed stage by stage to emulate human cognitive capacities for mathematics. I then analyse the two approaches in terms of their fit with philosophical theories of mathematical knowledge.

1. Introduction

The importance of artificial intelligence (AI) has been growing rapidly in many areas in the recent decades. Even within this long-term development, the enormous progress made in the past few years due to advances in machine learning applications based on multi-layered (i.e., *deep*) artificial neural network (ANN) architecture has been remarkable. Many problems that, for a long time, used to be considered exceedingly difficult for AI systems, such as translation and image recognition, are now routinely processed by them (see, e.g., Mitchell, 2019). Rule-based systems, often called the "good old-fashioned AI", did not disappear from AI research, of course, but the main thrust of the current golden period of AI research has undoubtedly come from machine learning systems run on deep neural networks.

In this development, mathematical AI has been something of an outlier. While machine learning systems are used regularly for a wide variety of tasks in other fields, in mathematics the standard AI tools are still rule-based systems. These include multi-purpose computing tool software like *Mathematica* and *Matlab*, as well as automated and interactive theorem proving software like *Isabelle, Mizar, Lean* and *E*.[1] For many mathematical purposes, the current generation of software is well-suited. The rule-based systems can be used efficiently and reliably to solve any problems for which there exists

[1] It should be noted to here that the *Mathematica* and *Matlab* platforms also have machine learning applications, but for mathematicians their standard use is still rule-based.

a general problem-solving algorithm (i.e., a finite set of rules for finding the solution). However, there are many areas of mathematics where such algorithms are not available. The most important of these is the proving of new theorems, which is a central activity of many professional mathematicians.

Strictly speaking, general algorithms for proving new theorems are possible. In the simplest scenario, such an algorithm takes a finite system of axioms as its input and generates theorems that follow logically from the axioms as the output. This kind of indiscriminatory output, however, is not particularly useful to mathematicians, given that most theorems in axiomatic systems are not interesting to mathematicians. Therefore, I submit that the relevant question is not whether AI systems can prove new theorems; it is whether they can provide as their output only (or at least to large extent) *interesting* theorems. This prompts the question: to whom should these theorems be *interesting*? This is not a trivial question: with high enough level, it is plausible that an AI system can develop its own standards of what interesting mathematical theorems are. However, here I focus on what I see as a more likely short-term goal of mathematical AI research, namely: acquiring proofs of new theorems that are interesting to *human* mathematicians.

I distinguish between two approaches to developing such *human-like artificial mathematical intelligence*. The first one I call the *bottom-up approach*, in which AI systems are developed to emulate human cognition already starting from very basic, non-linguistic levels. The second I call the *top-down approach*, which refers to training an AI system with mathematical data, i.e., existing proofs of theorems. Both approaches are present in contemporary research projects but, as I explain, they offer altogether different potential advantages, while also facing largely different challenges. Below, I present and analyse these advantages and challenges.

The paper is structured in five sections, as follows. In section 2, I present a summary of the use of AI applications in mathematics, both historically and currently. Then, in section 3, I outline the top-down account to developing human-like mathematical AI and the prospects of this research paradigm. In section 4, I focus on the bottom-up approach, reviewing the state of the art and potential future developments. Finally, in section 5, I analyse the philosophical importance of the two approaches, especially concerning their connection to path-dependency of development of mathematical cognition and knowledge.

2. Artificial intelligence in mathematics

While mathematics is currently not among the celebrated success stories of AI research, historically it has been a very important topic in the field. Indeed, some of the most famous examples in the history of machine solutions to cognitive tasks revolve around mathematics, starting from the designs of

Leonardo da Vinci and the first functional mechanical calculators for arithmetical operations introduced in the 17th century by Wilhelm Schickard and Blaise Pascal (Russell & Norvig, 2020, Section 1.2.1). Furthermore, in early AI research, proving mathematical theorems was a key aim. In fact, what is often called the first AI program[2] (see, e.g., Crevier, 1993), *The Logic Theorist* by Newell, Shaw and Simon (1957), proved theorems of mathematics, namely from Whitehead and Russell's *Principia Mathematica* (1910). *The Logic Theorist* proved theorems of that book (38 of the first 52 theorems in Chapter 2), in one case even providing a proof that was deemed to be more elegant than the original one (McCorduck & Cfe, 2004, p. 167).

After these early successes, mathematics remained to be a focus of AI research, and much progress has been made in many types of computer-based mathematical problem-solving. As mentioned in the introduction, software packages like *Matlab* and *Mathematica* have become standard tools for mathematical processing in many fields. For the most part, however, they are tools for *applied* mathematics. For "pure" mathematics, by which I here refer mainly to proving theorems, the situation is essentially different. That is not to say that AI applications cannot be important for research done by mathematicians working on theoretical proofs.[3] Currently they have a large selection of automated theorem prover software at their disposal, including (but not limited to) *Isabelle, Vampire, Prover9, Mizar, OTTER, Waldmeister, Lean* and *E*. However, the application and scope of such software is limited. Typically, they are fed a problem by a human user as input, consisting of a set of axioms (typically first-order formulas) and a conjecture (also a first-order formula). Then usually using first-order logic with equality, the software checks whether the conjecture follows from the axioms (Voronkov, 2003, p. 1607). Instead of a mere confirmation or disconfirmation, of course, the software should preferably also produce a humanly readable proof, in the case that the conjecture is confirmed as a theorem (ibid.).

Given the above explanation of the functioning of the software, it comes as no surprise that automated theorem provers are often also called *interactive theorem provers* and *proof assistants*. Their use is characterized by a constant interaction between the human and the software; instead of proving theorems autonomously, they are used as tools to assist humans in proving theorems. As such, they have become important tools for the mathematical community (Barendregt & Wiedijk, 2005). There is, for example, an on-going effort to formalize existing mathematics with the help of automated theorem provers, helping mathematicians check the validity of their proofs.[4]

[2] This claim is impossible to maintain, however, given that Strachey had already in 1951 presented a computer program that played checkers (draughts) (Strachey, 1952).

[3] To the best of my knowledge, no reliable data on the usage of AI tools among mathematicians is available.

[4] The most developed such resource currently is the *Mizar Mathematical Library* (http://www.mizar.org/library).

The importance of proof assistants and interactive theorem proving is likely to increase within mathematical practice as the software are developed further, but the question I am interested here is whether automated theorem proving software can function *autonomously*. Instead of checking humanly generated conjectures, could an AI application come up with new theorems without them being fed as input? As mentioned, in principle there is a simple way of doing that by the AI generating all logical consequences of the axioms. In practice this is of course impossible, given that most interesting mathematical systems are infinite. However, even for finite subsets of the theorems, the approach is unfeasible. Most theorems are likely to be completely uninteresting to human mathematicians and finding the uncommon, interesting ones in enormous inputs would probably be such a cumbersome task that it would defeat the entire purpose of using an automated theorem prover.

Therefore, for this kind of autonomous automated theorem proving, the AI application itself would need to possess ways of discriminating between interesting theorems and proofs and those that mathematicians consider to be trivial or otherwise uninteresting. There are some obvious cases in which the logical form of the theorem allows detecting it as trivial, such as symmetrical theorems of the form *A if and only if A*. Moreover, for proofs it is easy to agree on some heuristic criteria: for example, shorter proofs of a particular theorem are likely to be more interesting than longer proofs. Indeed, there have been efforts to find procedures for searching the shortest proof of a theorem (Fitelson & Wos, 2001; Kinyon, 2019; Veroff, 2001). However, such procedures are limited, in addition to not including any other criteria besides the length of the proof (for more, see (Pantsar, 2024b)). Therefore, it can be said that in the current state-of-the-art, very little progress has been made in terms of AI systems discriminating between theorems and proofs in terms of their potential interest to mathematicians (for more, see Pantsar, 2025a).

This situation is not surprising. After all, the current generation of automated theorem proving software consists of rule-based systems and, aside from some simple criteria like the ones described above, it is very difficult to formulate rules concerning what theorems and proofs are interesting. In fact, it has proven extremely challenging even to determine criteria for what mathematicians consider interesting. In assessing proofs, notions like “insightfulness” (Macbeth, 2012; Weber, 2010) and “beauty” (Johnson & Steinerberger, 2019; Rota, 1997) have been suggested, but they proved to be highly elusive concepts. Certainly, there is currently very little hope of capturing mathematical insightfulness or beauty in terms of rules that can be programmed into automated theorem provers.[5] For theorems, similar problems abound. The applicability of theorems both within mathematics and more widely in science is often

[5] The same goes for other similar suggestions, such as that of Thomas (2017), who proposed that being interesting should count itself as an aesthetic criterion in mathematics.

mentioned as criteria of being interesting (see, e.g., Lange, 2017). However, applicability does not seem to be necessary for a theorem to be interesting, given how some famous theorems are very specialized and purely theoretical.[6] In addition, applicability does not seem to be any easier to capture by rules than insightfulness and beauty are.

The current situation concerning discrimination between humanly interesting and uninteresting theorems and proofs is unlikely to change drastically when it comes to rule-based AI systems. That is because problem is fundamentally not a technological one; instead, it is about the difficulty of capturing rules governing the human mathematical practices in such discrimination. Progress can be made in understanding notions like mathematical insightfulness, beauty and applicability. Indeed, in the field of research called philosophy of mathematical practice, such questions play a central role (Mancosu, 2008). However, it seems unlikely that this progress can ever lead to the kind of formal rules that would be required to make them function as part of autonomous theorem provers.

For that reason, I want to analyse alternative approaches to automated theorem proving. As such, I reflect on the scenario of letting the AI detect patterns in what human mathematicians consider to be interesting, instead of trying to figure out rules governing such elusive notions as insightfulness, beauty or applicability. This type of approach has proven to be highly efficient and surprisingly accurate in fields like translation and image recognition. Could a similar development take place in the field of mathematics? In the next two sections, I present two potential approaches for developing such mathematical AI applications in machine learning systems based on artificial neural networks.

3. The top-down approach

Here I call the first approach to developing human-like mathematical AI *top-down*. The idea behind this approach is very similar to the way large language models work in translation and text generation. Currently the most famous such models are the different versions of the *GPT* (*Generative Pre-trained Transformer*) large language model developed by OpenAI, which are the basis for the ChatGPT chatbot.[7] As of the writing of this paper, the most recent GPT model is GPT-4o3, which is a multimodal model that accepts both text and image inputs (while producing textual outputs). It is an improvement on GPT-4 which, according to its developers, already ”exhibits human-level performance on various professional and academic benchmarks” (OpenAI, 2023). Interestingly for the present topic, this includes the SAT Math test, in which GPT-4 is reported to score 700/800 points, corresponding to the

[6] Fermat’s Last Theorem being a good example.

[7] But also the widely used translation tools like *DeepL and Google Translate* function based on large language models.

89th percentile in humans (OpenAI, 2023, p. 5). The SAT is mainly a multiple-choice test and as such not representative of mathematical practice. Its performance prompts the question whether a large model like GPT-4 could perform in a human-like fashion more generally in mathematical tasks.

The large language (and multimodal) models are created by deep artificial neural networks. In this type of neural network, the computer is trained with massive datasets. It then detects patterns in the training data, which it uses to predict most likely outputs corresponding to inputs, like the prompts presented to the ChatGPT chatbot (for technical details, see Aggarwal, 2023). The details of the training of GPT-4 are a trade secret, but we can assume that the training data includes examples of SAT math test questions. Therefore, its success in the test should not be overestimated. Indeed, ChatGPT can be notoriously unreliable in both numerical and logical tasks (see, e.g., Arkoudas, 2023). The only wider study on the topic at the time of writing this shows that GPT-4 is useful for undergraduate-level mathematics but fails at graduate-level difficulty (Frieder et al., 2023). This is nothing surprising for two main reasons. First, the GPT models are trained on internet texts, with the aim to generate human-like textual outputs. Therefore, it is not trained specifically for mathematical tasks. Second, the way the model works is fundamentally probabilistic, based on finding the most likely token to follow a string of tokens. This functioning is highly different from mathematical reasoning, which is based on the exact following of logical rules (Zvornicanin, 2023).

While these difficulties should not be downplayed, they both seem to be of the type where much more progress can be made by machine learning systems. Indeed, much progress has already been made in the new versions of ChatGPT. While the SAT math test should not be seen as a reliable indicator of mathematical ability, it is not insignificant that while its predecessor GPT-3.5 performed corresponding to the 70th human percentile in the test, GPT-4's performance corresponded to the 89th percentile (OpenAI, 2023, p. 5). Since details on the training data are not available, we cannot know whether this is due to development in the architecture of the model itself, or whether there was significant difference in the training data. The latter possibility is intriguing because the consequences of training GPT models specifically with mathematical material are not currently known.

It is tempting to hypothesize that the probabilistic architecture of the GPT models prevents them from acquiring reliable mathematical abilities also in future incarnations. However, it is important to remember that it was common for a long time to assume that successful text generation requires grasping rules of syntax (some, like (Chomsky et al., 2023), still do; see (Lee, 2023) for a different opinion). Yet the probabilistic large language (and multimodal) models have proven to perform up to very high standards without (supposedly) having such grasp. Could something similar happen in the case of mathematics? Could a machine learning system detect patterns in training data that would enable it to perform mathematical reasoning on similarly high levels? While this is an open question, some developments suggest that high-level mathematics may not be beyond the reach of deep neural networks. Progress with *neural theorem provers* gives reason for optimism in generating humanly interesting proofs and theorems. Unlike a rule-based system, a neural theorem prover can be trained with human-created proofs to teach strategies for completing proof steps (for an introduction, see Jenson, 2023). This approach was used recently with promising success, including building a library of "skills" to augment the theorem proving capacity of large language models (Lample et al., 2022; H. Wang et al., 2023). Similar machine learning applications have also demonstrated success in related tasks, such as premise selection (the problem of finding mathematical statements that are relevant for proving a particular conjecture) (M. Wang et al., 2017).

However, this "top-down" approach to developing human-like artificial mathematical intelligence has some fundamental problems. One important problem is the *epistemic opacity* of the deep neural networks (Durán & Formanek, 2018; Humphreys, 2009). Machine learning systems run on neural networks can be highly predictive, but due to their architecture and the sheer number of parameters it is impossible to trace *how* they end up with a certain output (Kay, 2018). This is known as the "black box" problem in the literature (Russell & Norvig, 2020, Section 19.9.4). Often the only data we get from machine learning systems is its output, which raises questions about their reliability. How can we know that the AI system followed valid reasoning in proving a theorem? While the growing *explainable AI (XAI)* research area focuses on this question, the problem cannot be expected to disappear (Doran et al., 2017; Holzinger, 2018; Thompson, 2021).

This is a potential problem also with one system recently reported to have had great success in solving mathematical problems, OpenMind's *o1* (OpenAI, 2024). That system was designed to iterate on the response before providing the actual output ("spend more time thinking" is the developers' remarkably unhelpful and misleading description). Fundamentally, however, *o1* functions based on the same principles as ChatGPT. As such, it remains vulnerable to mistakes and faces the black box problem. In this sense, however, two other applications introduced in 2024 would appear to have much more promise. These are DeepMind's applications *AlphaGeometry* (DeepMind, 2024b) and *AlphaProof* (DeepMind, 2024a) and Harmonic's *Aristotle* (Weinberg, 2024). Instead of being based on fundamentally on large language models, these AI systems are based on a hybrid, *neuro-symbolic* architecture. In this architecture, a large language model is first pre-trained on mathematical problems, which is then used to generate possible proof steps in solving the problem. These steps are then processed in the rule-based system theorem proving system *Lean* and successful steps are used to reinforce the model (DeepMind, 2024a).

So far, the hybrid systems have reached success in solving problems of the International Mathematical Olympiad (reaching a silver-medal level performance), but it is feasible that similar architecture could be used also in theorem proving. If possible, doing so could bypass the problem mentioned in the previous section concerning the difficulty of formalizing rules for interesting theorems and proofs. If the AI system were trained with datasets consisting of the kind of theorems that humans find interesting, it could detect implicit patterns that are not necessarily known to human mathematicians. On this basis, it could develop an ability to generate new theorems and proofs that follow the patterns in the training data. If an AI system could do both of these things, its great advantage over the current generation of theorem provers would be that it would be able to discriminate between interesting and trivial (or otherwise uninteresting) AI-produced theorems and proofs. Presently, this kind of mathematical AI remains in the realm of science fiction, but with the rapid growth in AI development – especially with the introduction of the hybrid neuro-symbolic systems – such a scenario no longer seems unrealistic.

Importantly, for the hybrid neuro-symbolic systems, consequences of the black box problem are not as serious as in other fields. While the large language model part of the system is opaque, the rule-based part of the system can ensure that logical rules are followed properly. Moreover, automated theorem proving may not be generally as vulnerable to the black box problem as other fields of generative AI. Instead of providing only a theorem as the output, the aim of developing automated theorem provers is to also provide the accompanying proof. This proof can be checked by rule-based AI systems – whether integrated in the system or not – but also by human mathematicians. In this respect, the AI-generated new theorems and proofs would not be essentially different from humanly generated ones. Assessing their correctness and determining their usefulness would ultimately be the task of the *human* mathematical community, perhaps using rule-based AI tools in the process.

There is, however, one potentially serious problem specific to the top-down approach. Deep neural networks are typically trained with enormous datasets, and it is possible that sufficiently large datasets of humanly interesting mathematical theorems and proofs are not available. Certainly, the theorems that mathematicians consider interesting are not in the order of *millions*. In the case of *AlphaProof*, a dataset of one million informal problems was used to create the training material of 100 million formal problems (DeepMind, 2024a). For theorems, however, this kind of process may not be possible. This difficulty leaves two options for the developers of theorem proving artificial networks. Either they need to find ways to train the system with relatively small datasets, or they need to find a way to generate datasets of interesting theorems and proofs. This latter option may sound odd: after all, it would amount to generating datasets of humanly interesting and proofs that humans have *not* established as interesting. Difficult as that seems, it can still be feasible. Actual published theorems and proofs could be used as models to create such datasets, which would be based on structural similarities.

We are likely to see rapid development in the top-down approach to developing human-like mathematical AI. Whereas in general the rate of progress in generative AI seems to be slowing down, in mathematical AI there is likely to be much more room for improvement. This is because the strength of large language models is closely connected to the amount of training data that was used in creating them. But adding new training data will become increasingly difficult. However, in the case of mathematics, as detailed above, this can still be done. If there is a way to generate new mathematical training data, we may see quick progress in mathematical AI, feasibly also in the case of theorem proving. Therefore, the top-down approach should be taken seriously by the mathematical community already at this point as potentially transformative to mathematical practice.[8]

4. The bottom-up approach

The top-down approach to developing human-like artificial mathematical intelligence is characterized by it being trained only with mathematical content. In that approach, researchers are not concerned about the question whether the functioning of the AI system otherwise emulates human thinking. Indeed, due to the black box problem, it is impossible to determine this. The aim of the top-down approach is therefore simply to create an AI system that can provide mathematically interesting output. This is comparable, for example, to AI translation tools, which are meant to provide accurate translations regardless of the way that is achieved.

Another approach to artificial mathematical intelligence is to emulate human cognition in a more fundamental way, encompassing lower-level cognitive processes. I call this the *bottom-up approach*. Unlike in the top-down approach, in the bottom-up approach the rules based on which the AI system functions are important. The ideal of the bottom-up approach is enabling emulation on all levels in the development of human mathematical cognition, creating an AI system that processes mathematics in an essentially human-like manner, but with the greater computational power of computers. Such an AI system would develop a human-like "sense" of what is interesting mathematics and what is not. If successful, the AI could use this capacity to provide humanly interesting theorems and proofs as its output. Compared to the top-down approach described in the previous section, this approach could then achieve similar capacity in recognising interesting mathematical content, but based on an different development principles.

It should be noted that this kind of approach to mathematical AI can be used for two different purposes, which are not necessarily connected. Given my focus in this paper, I will concentrate on AI systems that

[8] If this happens, it will cause important ethical issues about, e.g., authorship. For an analysis of this issue, see (Pantsar, 2025b).

are developed to assist mathematical progress. But human-like mathematical artificial intelligence could also be developed to understand human mathematical cognition better. The principle behind this approach is that if we can emulate some human cognitive capacity by a computer, it can help us explain the functioning of this capacity in the human brain and body (Pantsar, 2023).[9] Much of the research involved in what I call here the bottom-up approach seems to be mainly focused on explaining the human capacities in the early development of mathematical cognition (e.g., Di Nuovo & McClelland, 2019; Fang et al., 2018; Stoianov & Zorzi, 2012; for an overview, see Pantsar, 2023). However, this research is also relevant for the possibility of developing artificial human-like mathematical intelligence, on which I focus in the remainder of this section.

Regarding numbers, the educationally fundamental area of mathematics is the arithmetic of natural numbers. While this is not necessarily *conceptually* the case – numbers can be defined in terms of sets, for example (see, e.g., Enderton, 1977) – this fundamentality is generally extended also to the development of mathematical cognition (see, e.g., Lakoff & Núñez, 2000). Hence, in the bottom-up approach to developing mathematical AI, a feasible starting point is to first emulate arithmetical cognition. According to the current understanding, however, the development of arithmetic is based on evolutionarily developed quantitative abilities, called either *proto-arithmetical* (Pantsar, 2014) or *quantical* (Núñez, 2017) in the literature. Therefore, to follow the human developmental trajectory in developing artificial mathematical intelligence, it is necessary to start already from the proto-arithmetical level.

Typically, it is thought that humans possess two proto-arithmetical abilities (for an overview, see (Pantsar, 2024a)). *Subitizing* refers to the ability to determine the amount of observed objects without counting. This ability is exact but only works up to three or four objects (Knops, 2020). For larger collections, humans use an *estimating* ability that becomes increasingly inaccurate as the estimated collections of objects become larger (Dehaene, 2011). Both abilities are present already in infants and they are also possessed by many non-human animals (Dehaene, 2011; Starkey & Cooper, 1980; Xu & Spelke, 2000). While there is some debate over the cognitive basis of the proto-arithmetical abilities, the most common explanation is that they are due to *cognitive core systems*, i.e., innate systems that have developed through processes of biological evolution for specific cognitive purposes (Spelke, 2000). The subitizing ability is associated with the *object tracking system* (OTS) and the estimating ability with the *approximate number system* (ANS) (Carey, 2009; Hyde, 2011; Pantsar, 2019). In the context of developing human-like artificial arithmetical intelligence, the question to tackle is then whether we could, or should, emulate the proto-arithmetical abilities in AI systems.

[9] Mitchell (2019) calls these types of approaches the ”scientific side” of AI research, contrasting them with the practical side of engineering AI tools.

During the last decade or so, several researchers have worked on emulating the proto-arithmetical abilities. One important pioneering experiment in this research direction was reported in (Stoianov & Zorzi, 2012). They presented a deep artificial neural network with two-dimensional images with different sizes and numbers of dots, which is a standard method for studying pre-symbolic numerical abilities in humans (Dehaene, 2011; Xu & Spelke, 2000). This was done by unsupervised learning, so that the network was not trained to focus on any specific aspect of the input. Stoianov and Zorzi found out that the system learned to perform numerosity comparison tasks with similar behavioural signatures to those of the proto-arithmetical abilities of humans and non-human animals. In an interesting further result, the response profiles of the emergent "numerosity detectors" in the network resembled those reported in the lateral intraparietal area of macaque brains (Roitman et al., 2007; Stoianov & Zorzi, 2012). This type of research suggests that we can emulate early human non-symbolic numerical abilities with an AI, thus giving reason for optimism for the bottom-up approach (McClelland et al., 2016).

Further reasons for optimism have emerged from subsequent research. Testolin and colleagues (2020) report that a neural network could also develop similar numerical ability after being trained by a dataset of "natural" visual stimuli derived from, among other things, groups of animals.[10] The reported learning trajectories are highly similar to those reported in longitudinal studies of human proto-arithmetical abilities, and the final competence of the neural network approximated that of human adults (Halberda & Feigenson, 2008; Piazza et al., 2010). Such results obviously provide interesting material for the study of human arithmetical cognition. They may suggest that the proto-arithmetical abilities do not need cognitive core systems to develop, thus challenging the dominant current hypothesis on the subject. This stands out as particularly interesting in light of a study by Chen and colleagues (2018), according to which the data from artificial neural networks only conforms to the human proto-arithmetical abilities for numerosities larger than four. Since that is the limit of the OTS and the subitizing ability, it could imply that the ANS-hypothesis is not required to explain the development of proto-arithmetical abilities (Pantsar, 2023). This research is still in early stages, though, and sharp conclusions cannot be yet made. What needs to be assessed presently is whether such developments suggest a way forward in the bottom-up approach to developing human-like mathematical intelligence.

There are some early positive signs that such development is possible. While the mentioned experiments concern unsupervised learning in the AI system, for further development I direct my

[10] This dataset consisted of images with rectangular boxes indicating sizes and positions of objects in natural scenes (like the groups of animals), which were generated from computer vision data sets used in the PASCAL detection challenge (Everingham et al., 2010).

attention to supervised learning, i.e., machine learning where the system is trained to produce a desired output. This mirrors the ontogenetic cognitive development in humans: while proto-arithmetical abilities are innate, in order to develop proper arithmetical abilities humans need to acquire new concepts, rules and practices (Pantsar, 2019). Following this type of approach, Fang and colleagues (2018) used *teacher guided learning* to teach a neural network a counting procedure. In the experiment, two-dimensional blobs are given to the network as input. The guiding idea is that the network "touches" the blobs while connecting the procedure to numeral words, simulating how children learn to count by pointing to objects.[11] The teacher provided the correct counting procedure as the training data, but otherwise the neural networks were generic systems with no pre-trained ability with numerosities. The results show that after mastering the touching procedure, the network reached almost perfect rates in counting to six after 2,000 training trials. With more trials, it learned to count further (Fang et al., 2018).

Fang and colleagues' experiment intended to simulate the way human children learn to count, where gestures like pointing are advantageous (Alibali & DiRusso, 1999). Through another experiment, Di Nuovo and McClelland (2019) extended this approach to include embodied aspects in learning counting procedures. In the experiment, a humanoid robot with functional five-fingered "hands" was trained to use the fingers to represent spoken numerals. The AI received proprioceptive information from the robot hands, intended to emulate tactile and proprioceptive sensory input in humans. Their analysis showed that the proprioceptive information improved accuracy in recognizing spoken numeral words, established through the AI being faster in creating a uniform number line than a control AI system without the robot hand. Similar results were reported for a humanoid robot also in (Pecyna et al., 2020). These results have counterparts in the study of human numerical cognition where finger counting procedures have been shown to be advantageous for children in learning to count (Bender & Beller, 2012).

It needs to be remembered that learning to count is a very early stage in the development of arithmetical cognition. Indeed, in the stage of learning the counting procedure, children do not even possess number concepts (Davidson et al., 2012; Pantsar, 2021). Hence, the kinds of results reviewed above are a far cry from developing an AI that could even do basic arithmetic, let alone engage in sophisticated mathematical activity like proving theorems. However, the progress so far shows, at the very least, that human cognitive capacities connected to arithmetic can be simulated by AI systems.

[11] Here "touching" means the network having the position of a point (a coordinate pair on the display) as part of its output, which was treated as both the centre of gaze and the location it was touching on the display (Fang et al., 2018).

This progress gives hope that we could train an AI system in a cumulative manner that mirrors human mathematical development. We could first (through unsupervised learning) train it to simulate human proto-arithmetical capacities, as done by (Stoianov & Zorzi, 2012) and (Testolin et al., 2020). Then we could teach the system to count, as done by (Fang et al., 2018) and (Di Nuovo & McClelland, 2019). Once the system masters the counting procedure without limits, we can move on to teaching arithmetical operations. Then, the mathematical "education" of the AI system could be expanded to include negative numbers, rational numbers, real numbers, etc., following the educational paths in place for human children. In the final stages of this kind of project, the AI system could then learn formal mathematics and be able to prove theorems. If the bottom-up approach works according to the optimist view, the AI system at that point would have implicit rules for discriminating humanly interesting theorems and proofs from uninteresting ones. After all, at every stage it was trained in a similar way to how human mathematicians are trained.

Is such a scenario realistic? It is difficult to predict, given that the research is currently in very early stages. Still, even if we could not reach the level where the AI system compares to human mathematicians in its ability, the bottom-up approach presents many advantages. As mentioned, it can help to better explain human cognitive processes involved in mathematics. Consequently, an AI with some human-like mathematical abilities would be a valuable tool also for research in mathematics education. Experimenting on new educational practices with human students is an unpredictable process that takes a lot of time. In addition, it may be risky or even unethical, given that educational practices may cause disadvantages for students in their entire educational trajectory. But with AI systems there would be no such problems. We could experiment on new practices much more quickly without fear of damaging anybody's academic (and general) future. Therefore, the bottom-up approach has advantages and potential applications even if it could not be completed to the top levels of human-like mathematics.

5. The philosophy of artificial mathematical intelligence

Above I have described two contrasting paradigms for developing human-like artificial mathematical intelligence: the top-down and the bottom-up approaches. The two approaches have many technical differences, with important consequences for their feasibility. In this section, I focus on the philosophical differences between these approaches. Specifically, I discuss how each approach relates to philosophical accounts of mathematical cognition and the epistemology of mathematics. I propose that the best way to do this is by focusing on the ways in which mathematical knowledge develops both in terms of individual ontogeny and population-level phylogeny and cultural history, as well as the importance for that development in the philosophy of mathematics.

In Platonist philosophy of mathematics, which has been a dominant tradition in the philosophy of mathematics for the most of the history of the discipline, mathematical truths are thought to concern mind-independent mathematical objects and their relations (Linnebo, 2018). Since Platonic mathematical objects are abstract (i.e., non-physical), knowledge about them is acquired by reason and recollection (Plato, *The Republic*). For an AI system to acquire mathematical knowledge, it would need to be able to reason in a proper way, i.e., similarly to human reasoning. In the top-down approach, this is possible by the AI detecting patterns in humanly produced mathematical work. Such an AI system can establish rules for creating new mathematical content based on these patterns. In the bottom-up approach, this reasoning ability could emerge from a gradual development and training of the AI system, starting from the proto-arithmetical abilities, mirroring the way in which humans gradually acquire the ability for mathematical reasoning during their cognitive development and educational trajectory. In this sense, Platonist mathematics seems to be an equally good (or bad) fit with both approaches to mathematical AI. If mathematical knowledge is considered mind-independent and objective, the developmental trajectory of intelligence (whether biological or artificial) would only seem to matter in the practical sense that some trajectories are better for acquiring the reasoning ability and relevant knowledge. There would not seem to be, *prima facie* at least, any reason why this kind of ability and knowledge could not in principle be acquired by artificial systems.

Is the matter different if mathematical knowledge is considered a human creation and therefore mind-*dependent*? Such approaches have been presented by several researchers, ranging from social constructivist accounts (Cole, 2013, 2015; Feferman, 2009) to conventionalist accounts (Warren, 2020), as well as accounts in which mathematical knowledge is seen as (at least partly) determined by our innate cognitive architecture, including the proto-arithmetical abilities (Lakoff & Núñez, 2000; Pantsar, 2014). While the accounts differ in important ways (see (Pantsar, 2024a) for an overview), they share the central characteristic that the subject matter of mathematical knowledge is thought to be determined by human cognitive and cultural practices.

What can be said about the prospect of developing artificial mathematical intelligence in accordance with such mind-dependent accounts? At first glance, it may seem that they are a better fit with the bottom-up approach. After all, if we manage to emulate artificially the different stages of human mathematical development in individual ontogeny, the developmental trajectory of the AI mirrors (at least to some degree) the developmental trajectory of human subjects. This may or may not also mirror the way in which mathematical knowledge has developed on the phylogenetic and cultural level. However, this matter is irrelevant for the present question: if the bottom-up approach is feasible, it is enough for the AI system to emulate the ontogenetic trajectory to reach human-like mathematical intelligence.

Now the question is, does a system (whether biological or artificial) need to follow the human ontogenetic trajectory and if so, to what degree? To answer this question, it is important to recognize that there is no single trajectory of the development of human mathematical cognition. On the historical level, arithmetic, for example, was developed independently by several cultures, with differences both in characteristics and applications (Ifrah, 1998). Similarly, there are important differences in the modern practices in learning arithmetic in different cultures, ranging from structural differences in numeral word systems (Pantsar & Stjernfelt, 2025) to different cognitive tools (e.g., pen and paper, abacus) (Fabry & Pantsar, 2021; Pantsar, 2019). Cases like that of Srinivasa Ramanujan show that, also on the higher levels of mathematical cognition, it is possible to reach expert ability without following a standard path of formal education.[12]

These considerations corroborate that mathematical knowledge is not necessarily tightly connected to a particular path of cognitive development and education. The key question for the present topic is then whether it could nevertheless be connected to such paths in some important way. Could we simply bypass the different stages of human cognitive development and train an AI system exclusively with higher mathematical content, as in the top-down approach? Or are some aspects of human cognitive development so integral to the development of mathematical ability that they cannot be bypassed? If that were the case, then only the bottom-up approach could potentially reach human-like artificial mathematical intelligence.

These questions are open as of writing this paper. Perhaps human mathematical reasoning has an element – for a lack of better word, let us call it *intuition* – that cannot be reconstructed only from mathematical articles. But it should also be remembered that for a long time it was thought that successful translation requires a similar component, namely *understanding*. Yet contemporary large language models have proven to work very well for translation without (as far we as we know) possessing any kind of understanding of the content that they are translating. Mathematical content could be similar: intuition and understanding may be important for human mathematicians, but that is due to our particular cognitive development and enculturation (Pantsar, 2019, 2024a). For an AI with enormous computational power – and this capacity will continue to increase greatly – there may be other ways to come up with the same output. As of now, the top-down and bottom-up approaches both seem to enable the development of human-like mathematical intelligence. This does not change based on whether we understand the subject matter of mathematics as mind-independent (as

[12] The Indian-born Ramanujan was a largely self-taught mathematician whose work was innovative and important in several fields of mathematics (Kanigel, 1991).

Platonists do) or dependent on human cognition and practices (as social constructivists, for example, do).

Conclusion

In this paper, I have analysed two different approaches to developing human-like artificial mathematical intelligence, using generating humanly interesting new theorems and proofs as the relevant target phenomenon. In the top-down approach, an AI system (e.g., a deep artificial neural network) is trained with mathematical material, i.e., articles and other texts containing humanly produced proofs and other such mathematical content. In a similar manner to the large language models behind many recent AI success stories, the AI system would then detect patterns in the content that would allow it to produce new mathematics that is humanly interesting. In the bottom-up approach, the AI system is developed by emulating human cognitive development, stage by stage. This starts already from human proto-arithmetical abilities before moving on to emulating educational trajectories in supervised learning processes. In this approach, the development of the AI system would be designed to follow the path of human cognitive development, with the ultimate aim of developing human-like ability to separate interesting and uninteresting mathematical content. When trained all the way to the highest level of mathematics, the AI system could then (potentially) produce new humanly interesting mathematics.

I have shown that from a philosophical point of view, both approaches are compatible with different ways of understanding mathematical knowledge. Therefore, the crucial questions concern the feasibility of the practical project of developing mathematical AI. If both approaches were in practice feasible, the top-down one would carry many important advantages. Perhaps most importantly, compared to the bottom-up approach it promises a much faster process of training the AI – and with the recent success of AlphaProof, AlphaGeometry 2 and Harmonic, we are already seeing this happen for mathematical problem solving. But can we bypass human developmental stages from the training process and simply train the AI with higher level mathematical material? This kind of approach to AI has proven to be very successful for translation and other natural language processing. It remains to be seen whether it can be successful also for the purpose of creating new mathematics.

References

Aggarwal, C. (2023). Deep Learning: Principles and Training Algorithms. In C. C. Aggarwal (Ed.), *Neural Networks and Deep Learning: A Textbook* (pp. 119–163). Springer International Publishing. https://doi.org/10.1007/978-3-031-29642-0_4

Alibali, M. W., & DiRusso, A. A. (1999). The function of gesture in learning to count: More than keeping track. *Cognitive Development*, *14*(1), Article 1.

Arkoudas, K. (2023). ChatGPT is no Stochastic Parrot. But it also Claims that 1 is Greater than 1. *Philosophy & Technology*, *36*(3), 54. https://doi.org/10.1007/s13347-023-00619-6

Barendregt, H., & Wiedijk, F. (2005). The challenge of computer mathematics. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, *363*(1835), Article 1835. https://doi.org/10.1098/rsta.2005.1650

Bender, A., & Beller, S. (2012). Nature and culture of finger counting: Diversity and representational effects of an embodied cognitive tool. *Cognition*, *124*(2), Article 2. https://doi.org/10.1016/j.cognition.2012.05.005

Carey, S. (2009). *The origin of concepts*. Oxford University Press.

Chen, S., Zhou, Z., Fang, M., & McClelland, J. (2018). Can generic neural networks estimate numerosity like humans? *CogSci*.

Chomsky, N., Roberts, I., & Watumull, J. (2023, March 8). Opinion | Noam Chomsky: The False Promise of ChatGPT. *The New York Times*. https://www.nytimes.com/2023/03/08/opinion/noam-chomsky-chatgpt-ai.html

Cole, J. C. (2013). Towards an Institutional Account of the Objectivity, Necessity, and Atemporality of Mathematics†. *Philosophia Mathematica*, *21*(1), Article 1. https://doi.org/10.1093/philmat/nks019

Cole, J. C. (2015). Social Construction, Mathematics, and the Collective Imposition of Function onto Reality. *Erkenntnis*, *80*(6), Article 6. https://doi.org/10.1007/s10670-014-9708-8

Crevier, D. (1993). *Ai: The Tumultuous History Of The Search For Artificial Intelligence* (First Edition). Basic Books.

Davidson, K., Eng, K., & Barner, D. (2012). Does learning to count involve a semantic induction? *Cognition*, *123*, 162–173.

DeepMind. (2024a, October 9). *AI achieves silver-medal standard solving International Mathematical Olympiad problems*. Google DeepMind. https://deepmind.google/discover/blog/ai-solves-imo-problems-at-silver-medal-level/

DeepMind. (2024b, October 9). *AlphaGeometry: An Olympiad-level AI system for geometry*. Google DeepMind. https://deepmind.google/discover/blog/alphageometry-an-olympiad-level-ai-system-for-geometry/

Dehaene, S. (2011). *The Number Sense: How the Mind Creates Mathematics, Revised and Updated Edition* (Revised, Updated ed. edition). Oxford University Press.

Di Nuovo, A., & McClelland, J. L. (2019). Developing the knowledge of number digits in a child-like robot. *Nature Machine Intelligence*, *1*(12), Article 12.

Doran, D., Schulz, S., & Besold, T. R. (2017). What does explainable AI really mean? A new conceptualization of perspectives. *arXiv Preprint arXiv:1710.00794*.

Durán, J. M., & Formanek, N. (2018). Grounds for Trust: Essential Epistemic Opacity and Computational Reliabilism. *Minds and Machines*, *28*(4), Article 4. https://doi.org/10.1007/s11023-018-9481-6

Enderton, H. B. (1977). *Elements of set theory*. Academic Press.

Everingham, M., Van Gool, L., Williams, C. K., Winn, J., & Zisserman, A. (2010). The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, *88*(2), Article 2.

Fabry, R. E., & Pantsar, M. (2021). A fresh look at research strategies in computational cognitive science: The case of enculturated mathematical problem solving. *Synthese*, *198*(4), Article 4. https://doi.org/10.1007/s11229-019-02276-9

Fang, M., Zhou, Z., Chen, S., & McClelland, J. (2018). Can a recurrent neural network learn to count things? *CogSci*.

Feferman, S. (2009). Conceptions of the continuum. *Intellectica*, *51*(1), Article 1.

Fitelson, B., & Wos, L. (2001). Finding Missing Proofs with Automated Reasoning. *Studia Logica: An International Journal for Symbolic Logic*, *68*(3), Article 3.

Frieder, S., Pinchetti, L., Chevalier, A., Griffiths, R.-R., Salvatori, T., Lukasiewicz, T., Petersen, P. C., & Berner, J. (2023). *Mathematical Capabilities of ChatGPT* (arXiv:2301.13867). arXiv. https://doi.org/10.48550/arXiv.2301.13867

Halberda, J., & Feigenson, L. (2008). Developmental change in the acuity of the" Number Sense": The Approximate Number System in 3-, 4-, 5-, and 6-year-olds and adults. *Developmental Psychology*, *44*(5), Article 5.

Holzinger, A. (2018). From machine learning to explainable AI. *2018 World Symposium on Digital Intelligence for Systems and Machines (DISA)*, 55–66.

Humphreys, P. (2009). The philosophical novelty of computer simulation methods. *Synthese*, *169*(3), Article 3. https://doi.org/10.1007/s11229-008-9435-2

Hyde, D. C. (2011). Two systems of non-symbolic numerical cognition. *Frontiers in Human Neuroscience*, *5*, 150.

Ifrah, G. (1998). *The universal history of numbers: From prehistory to the invention of the computer*. Harville Press.

Jenson, D. (2023, May 16). Automated Theorem Proving with Graph Neural Networks. *Stanford CS224W GraphML Tutorials*. https://medium.com/stanford-cs224w/automated-theorem-proving-with-graph-neural-networks-49c091024f81

Johnson, S. G. B., & Steinerberger, S. (2019). Intuitions about mathematical beauty: A case study in the aesthetic experience of ideas. *Cognition*, *189*, 242–259. https://doi.org/10.1016/j.cognition.2019.04.008

Kanigel, R. (1991). *The Man Who Knew Infinity: A Life of the Genius Ramanujan*. Scribner.

Kay, K. N. (2018). Principles for models of neural information processing. *NeuroImage*, *180*, 101–109. https://doi.org/10.1016/j.neuroimage.2017.08.016

Kinyon, M. (2019). Proof simplification and automated theorem proving. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, *377*(2140), Article 2140. https://doi.org/10.1098/rsta.2018.0034

Knops, A. (2020). *Numerical cognition. The basics*. Routledge.

Lakoff, G., & Núñez, R. (2000). *Where mathematics comes from*. Basic Books.

Lample, G., Lachaux, M.-A., Lavril, T., Martinet, X., Hayat, A., Ebner, G., Rodriguez, A., & Lacroix, T. (2022). *HyperTree Proof Search for Neural Theorem Proving* (arXiv:2205.11491). arXiv. https://doi.org/10.48550/arXiv.2205.11491

Lange, M. (2017). *Because without cause: Non-causal explanations in science and mathematics*. Oxford University Press.

Lee, E. (2023). *Is ChatGPT a False Promise?* Berkeley. https://news.berkeley.edu/2023/03/19/is-chatgpt-a-false-promise

Linnebo, Ø. (2018). Platonism in the Philosophy of Mathematics. In E. Zalta (Ed.), *The Stanford Encyclopedia of Philosophy*. https://plato.stanford.edu/archives/spr2018/entries/platonism-mathematics.

Macbeth, D. (2012). Proof and Understanding in Mathematical Practice. *Philosophia Scientiæ. Travaux d'histoire et de Philosophie Des Sciences*, *16–1*, Article 16–1. https://doi.org/10.4000/philosophiascientiae.712

Mancosu, P. (Ed.). (2008). *The Philosophy of Mathematical Practice* (1st edition). Oxford University Press.

McClelland, J. L., Mickey, K., Hansen, S., Yuan, A., & Lu, Q. (2016). A parallel-distributed processing approach to mathematical cognition. *Manuscript, Stanford University*.

McCorduck, P., & Cfe, C. (2004). *Machines Who Think: A Personal Inquiry into the History and Prospects of Artificial Intelligence* (2nd edition). A K Peters/CRC Press.

Mitchell, M. (2019). *Artificial Intelligence: A Guide for Thinking Humans* (Illustrated edition). Farrar, Straus and Giroux.

Newell, A., Shaw, J. C., & Simon, H. A. (1957). Empirical explorations of the logic theory machine: A case study in heuristic. *Papers Presented at the February 26-28, 1957, Western Joint Computer Conference: Techniques for Reliability*, 218–230.

Núñez, R. E. (2017). Is there really an evolved capacity for number? *Trends in Cognitive Science*, *21*, 409–424.

OpenAI. (2023). *GPT-4 Technical Report*. https://cdn.openai.com/papers/gpt-4.pdf

OpenAI. (2024). *Introducing OpenAI o1*. https://openai.com/index/introducing-openai-o1-preview/

Pantsar, M. (2014). An empirically feasible approach to the epistemology of arithmetic. *Synthese*, *191*(17), Article 17. https://doi.org/10.1007/s11229-014-0526-y

Pantsar, M. (2019). The enculturated move from proto-arithmetic to arithmetic. *Frontiers in Psychology*, *10*, 1454.

Pantsar, M. (2021). Bootstrapping of Integer Concepts: The Stronger Deviant-Interpretation Challenge. *Synthese*, *199*(3–4), Article 3–4. https://doi.org/10.1007/s11229-021-03046-2

Pantsar, M. (2023). Developing artificial human-like arithmetical intelligence (and why). *Minds and Machines*, *33*, 379–296. https://doi.org/10.1007/s11023-023-09636-y

Pantsar, M. (2024a). *Numerical Cognition and the Epistemology of Arithmetic*. Cambridge University Press.

Pantsar, M. (2024b). Theorem proving in artificial neural networks: New frontiers in mathematical AI. *European Journal for Philosophy of Science*, *14*(1), 4. https://doi.org/10.1007/s13194-024-00569-6

Pantsar, M. (2025a). How to Recognize Artificial Mathematical Intelligence in Theorem Proving. *Topoi*. https://doi.org/10.1007/s11245-025-10164-w

Pantsar, M. (2025b). The need for ethical guidelines in mathematical research in the time of generative AI. *AI and Ethics*. https://doi.org/10.1007/s43681-025-00660-5

Pantsar, M., & Stjernfelt, F. (2025). How Numeral Words and Symbols Shape Arithmetical Cognition. *Cybernetics and Human Knowing*, *31*(3–4), 111–128.

Pecyna, L., Cangelosi, A., & Di Nuovo, A. (2020). A robot that counts like a child: A developmental model of counting and pointing. *Psychological Research*, 1–17.

Piazza, M., Facoetti, A., Trussardi, A. N., Berteletti, I., Conte, S., Lucangeli, D., Dehaene, S., & Zorzi, M. (2010). Developmental trajectory of number acuity reveals a severe impairment in developmental dyscalculia. *Cognition*, *116*(1), Article 1. https://doi.org/10.1016/j.cognition.2010.03.012

Plato. (1992). *The Republic* (G. M. A. Grube, Trans.; second). Hackett Publishing Company. Putnam.

Roitman, J. D., Brannon, E. M., & Platt, M. L. (2007). Monotonic Coding of Numerosity in Macaque Lateral Intraparietal Area. *PLOS Biology*, *5*(8), Article 8. https://doi.org/10.1371/journal.pbio.0050208

Rota, G.-C. (1997). The Phenomenology of Mathematical Beauty. *Synthese*, *111*(2), 171–182.

Russell, S., & Norvig, P. (2020). *Artificial Intelligence: A Modern Approach* (4th edition). Pearson.

Spelke, E. S. (2000). Core knowledge. *American Psychologist*, *55*(11), Article 11. https://doi.org/10.1037/0003-066X.55.11.1233

Starkey, P., & Cooper, R. G. (1980). Perception of numbers by human infants. *Science*, *210*(4473), Article 4473.

Stoianov, I., & Zorzi, M. (2012). Emergence of a'visual number sense'in hierarchical generative models. *Nature Neuroscience*, *15*(2), Article 2.

Strachey, C. S. (1952). Logical or non-mathematical programmes. *Proceedings of the 1952 ACM National Meeting (Toronto)*, 46–49. https://doi.org/10.1145/800259.808992

Testolin, A., Zou, W. Y., & McClelland, J. L. (2020). Numerosity discrimination in deep neural networks: Initial competence, developmental refinement and experience statistics. *Developmental Science*, *23*(5), Article 5.

Thomas, R. S. D. (2017). Beauty is not all there is to Aesthetics in Mathematics†. *Philosophia Mathematica*, *25*(1), 116–127. https://doi.org/10.1093/philmat/nkw019

Thompson, J. A. F. (2021). *Forms of explanation and understanding for neuroscience and artificial intelligence*. PsyArXiv. https://doi.org/10.31234/osf.io/5g3pn

Veroff, R. (2001). Finding Shortest Proofs: An Application of Linked Inference Rules. *Journal of Automated Reasoning*, *27*, 123–139.

Voronkov, A. (2003). Automated Reasoning: Past Story and New Trends. *IJCAI*.

Wang, H., Xin, H., Zheng, C., Li, L., Liu, Z., Cao, Q., Huang, Y., Xiong, J., Shi, H., Xie, E., Yin, J., Li, Z., Liao, H., & Liang, X. (2023). *LEGO-Prover: Neural Theorem Proving with Growing Libraries* (arXiv:2310.00656). arXiv. https://doi.org/10.48550/arXiv.2310.00656

Wang, M., Tang, Y., Wang, J., & Deng, J. (2017). Premise Selection for Theorem Proving by Deep Graph Embedding. *arXiv:1709.09994 [Cs]*. http://arxiv.org/abs/1709.09994

Warren, J. (2020). *Shadows of syntax: Revitalizing logical and mathematical conventionalism*. Oxford University Press.

Weber, K. (2010). Proofs that develop insight. *For the Learning of Mathematics*, *30*(1), Article 1.

Weinberg, J. (2024, July 11). *New AI Venture Focuses on Mathematical Reasoning—Daily Nous*. https://dailynous.com/2024/07/11/new-ai-venture-focuses-on-mathematical-reasoning/, https://dailynous.com/2024/07/11/new-ai-venture-focuses-on-mathematical-reasoning/

Whitehead, A. N., & Russell, B. (1910). *Principia Mathematica—Volumes 1-3*. Cambridge University Press.

Xu, F., & Spelke, E. S. (2000). Large number discrimination in 6-month-old infants. *Cognition*, *74*(1), Article 1. https://doi.org/10.1016/S0010-0277(99)00066-9

Zvornicanin, E. (2023, August 4). *Why Is ChatGPT Bad at Math? | Baeldung on Computer Science*. https://www.baeldung.com/cs/chatgpt-math-problems