>wired_

3RD ED.



Contents

.

11

21

INTERVIEW WITH RAVI HAMMOND : FROM MINECRAFT TO MICROSOFT

Artificial Intelligence V/S Natural Intelligence A Contrasting Comparison

AlphaGeometry: a theorem prover for Euclidean geometry

Contributors

Surya Sathyamurthy Bonsen Wakjira Jeevan Rajagopal Rahul Pejathaya

Special Thanks to Ravi Hammond



Created by Faculty of Information Technology Society (WIRED) https://wired.org.au

Released in good faith and for free under the CC BY-SA 4.0 licence. €0.00 \$0.00 £0.00

INTERVIEW WITH RAVI HAMMOND : FROM MINECRAFT TO MICROSOFT



Meet Ravi Hammond – a rising star in Artificial Intelligence from Adelaide. With a background in Computer Science and a Master's in Al Research from the University of Adelaide, he's set to begin his PhD at Oxford this October. Ravi's industry experience spans internships at Google, Microsoft, Palantir, and Atlassian, and he was part of the team that placed third in the global NASA Space Robotics Challenge. He's also the founder of RSP, a non-profit empowering students through free tech bootcamps. Ravi brings a rare blend of academic excellence, real-world expertise, and a passion for giving back.

What do Minecraft redstone, Old School RuneScape, Google internships, and an upcoming Oxford AI PhD have in common? They're all part of Ravi Hammond's unique trajectory. From a challenging start to academic life, Ravi forged a path into elite tech circles and now dedicates significant time to lifting others through his non-profit, RSP.

Read on as Ravi discusses his journey, the surprising origins of his drive, the importance of building genuine community, and his practical advice for turning lofty aspirations into reality in a world full of digital noise.



Can you introduce yourself Ravi?

Hi, I'm Ravi Hammond, I am starting a PHD in AI research at Oxford in October, and I basically have a history of doing Computer Science at Adelaide University and doing a masters here as well in AI research. Outside of university, I have interned at a lot of companies as a software engineer, machine learning engineer and site reliability engineer at companies such as Palantir, Microsoft, Google and Atlassian. And some other things; I competed in the NASA space robotics challenge where we came third in the world. Those were some really fun times; that was a six month project and we had a team of 40 people all working on the competition. And, in the last few years I have founded and been running a non-profit, free bootcamp where we train undergrads for free to get into big tech companies.

Can you tell us where you started from? I mean you have had a master's, a bachelors, a PHD, so how was your early uni life like? How did you get into computer science?

Yea um, so I started pretty rough actually to be honest. I didn't finish year 12 and I had a slightly uncomfortable teenage period where I wasn't focused and didn't care about academia. And I guess I got a lot of socialising out of my system in grade 11 and 12, and all my friends started graduating university and I realised

I had to do something with my life, and I jumped around a bit with courses. At the time, I was playing a lot of computer games so I liked computers. Actually, I started doing programming in Minecraft with redstone. I would do redstone circuitry, and I would learn the basics of logic gates and circuits. And I know it's a common path that some people in university do, to get into programming and I thought it was cool. And I got quite good at wiring up this electricity, you can call it, inside of Minecraft. And that made me think, I want to go to uni and do electrical engineering, and I started the process for that. Because I didn't finish year 12 I had to redo year 12 maths and year 12 physics. My university in Adelaide had a program to help me reach that gap, and I entered uni at 21.

But when I was doing that program, I started learning how to make a mod for minecraft for fun. And I realised quite quickly that what I loved the most wasn't the wiring up of circuits in Minecraft. The stuff I really loved was when I would go to work in real life and I would be thinking about it and solving the redstone problem in my head, and then going back and coding/building it in the game. And I realised that was actually what programming was, it was just you doing that. So pretty quickly I realised, hey this is what I actually love so I switched my stuff around and I entered a degree in Computer Science.

When I went in, I was a late student. I was about 22. I felt a bit of pressure; all my friends were working now, and I felt this intensity of wanting to do something and get there quickly and do something well. But I also have a bit of a perfectionist mentality, and a part of why I didn't finish year 12 was because I was playing a lot of online games like Runescape, and I guess I had taught myself in those games where if there was something that seemed impossible, I would create spreadsheets and plan how to achieve those things in that game. I realised that I could do things that felt impossible by planning/spending months of my time in those games and eventually achieving those big feats.

And I definitely took that to uni. I knew from experience, from those computer games, that if you just plan it out, and apply it and stick to it and persevere with your little checklist of things you want to do, you know, I would tell people that I was treating myself like a computer game character and I'm just going to level myself up in a grindy type of way. So that was my mentality going into university, and you know I was putting a lot of effort into the first year of uni, grinding things out as fast as I could.

I would attend a lot of club activities as well, and I met some guys who were in the year above me who were interning at Google and Facebook and Microsoft., And that blew my mind completely, because at the time I just thought that maybe I could start a little programming school for kids. Because I was thinking, what would be useful for people and what I would want, and that is the only thing I could think of. But then when I met those guys, and they said oh I'm going to go to these big companies, that

suddenly became "impossible my goal"/"computer game goal. That sounds like an impossible goal, it felt just as impossible as some of those achievements from those computer games. So I kind of pivoted myself pretty quickly, I hung out with those guys a lot and I ended up being best mates with them. We would gym together and programming each problems give while gyming, it was an intensive group who were highly motivated to get into these big companies. I learnt from them, and that's what created that stream of me interning at companies throughout my undergraduate degree.

So that's really unexpected, usually when you hear stories like this, the person excels at every step of their way. It's really nice to hear someone who actually had struggles in their life and in their early years, and found their way through something like games as well. It is really interesting.

Is a computer science degree valuable or useful that much?

Long story short, to give you an actual answer, it doesn't matter so much because you are going to be doing your degree, and to do well in a degree you have to reach out and do other things that aren't in the degree, otherwise you can't get a good job. That's just a fact. Yes, you can roll the dice and get a good job, I'm not saying that can't happen, but statistically it is harder to get a really good job if you just do what the degree says, and not other things. Whether that's doing a Ravi's Study Program; if that's the case you don't even need to do computer science you can be a maths student. All companies want to see is that you are doing a degree, they don't care what the name of that degree is, unless it's like a company that requires you to be a qualified engineer or some other field. But maths/computer science/electrical engineering is all kind of blurry. People get a job in one field when their degree is in another field. As long as you're in that bucket, mathematics, electrical engineering, computer science; as long as the projects you do outside university or things you do outside university help you get that job you want, you'll get a job regardless.

I actually have a question on a tangent, do you prefer old school runescape, or the modern one?

Old School Runescape; and the reason for that is because I learnt the value of grinding because of how hard Runescape was. And that has transformed me into the person who grinds in their everyday life and gets good outcomes for my own career. And that's the most important thing I got out of runescape, so I prefer old school runescape for sure.

You've acknowledged the importance of community support in your success, what I'm referring to of course is your citizen of the year speech. How do you foster a similar sense of community among your students/colleagues wit in RSP or your day to day life? I think the most important thing for you to be a spearhead, or for anyone, myself, or anybody to be a spearhead, in community fostering is: for a long time, for most of it you need to be willing to give more than you get back. If we want to create a world where people are all giving to each other, then someone needs to lead by example and do the giving. And that's giving advice, time, or anything required, and basically the most important thing I love about my community I have created is that everyone puts in so much time into helping each other. That's because I initially ran the program myself and gave everyone hundreds and hundreds of hours of my time, and those students wanted to give back. And you know the motto I have is that, "This program will be free, it is always going to be free, and the way you can pay me back is to pay it forward and give it to the next kids". It's like a breath of fresh air, it's like you aren't in a university where people are trying to take your money, everyone just knows that everyone is here purely to help. It's the most important thing i do for my students, I have invested a lot of time into this community. Were a few other questions in there? You were also asking about how important it is?



When you are starting off, and you are in a new committee, how do you foster that sense of comradery with the people you work with, and how do you in turn, build that culture of paying it forward to the next generation. The kick-start of it is putting in your own time, but with a set and collective of people, how does that translate?

Yea of course. It's sometimes hard to think of your co-curricular activities as something not purely for the resume. A lot of people get involved in societies because of that, or maybe some percentage of their mindset is like that. I think the first thing that needs to be done in this community, is in four years time, when we're gone, how will we make the community the best it can be? What can we do for this community that isn't going to help me, but help people 5 years from now. For it to be genuine, for people who want to pay it forward, people need to know and 100% believe that you are doing it for no reason other than to help them. And that's a part of why people want to pay it forward, because they see that we are doing it, and they want to do it too.

I'm talking about a lot of roses and happy stuff here, but there's also a strictness to it. In my program, we do rule the program with an iron fist. So people have to do the work or they're out. That runs the risk of upsetting people and so. On top of wanting to be nice to people, we want to be working with people who actually really want it and people who respect the

program. We have set the rules to be strict, only 1/3 or 1/4 of the people actually finishes the program. That creates this feeling of respect, where people respect the program and the results. They know it's not an easy thing, and people want to do things that are hard, we had 500 applications last year, and everyone knows how crazy hard it is, but people want to do things that are hard. They want to challenge themselves. So when you guys are creating this community and on the drawing board, there needs to be at least one person that needs to make sure it is being run with an iron first. Because I have been trying to teach people for years, I've been trying to give people advice for years, and all throughout my undergraduate, I had friends, family I tried to give advice like: "here's what you can do to get into Google", and the common thread is me trying to help people. In the end, it's so much work that people just won't do it, and that's why having the program where you do it or you're out works. That was the only solution I found after trying for many many years.

How did you go about refining this accountability model? Was it if you didn't do this set amount of work you're out? Or was it that you don't have the dedication or passion? How did you come across this conclusion? Was it a gradual decision or something you decided early on? How did that go? Initially, I was having coffee with 10 people at a time, and I decided to put them into the program and that was the initiation for RSP. I thought about it alot, and I already knew that I don't want to be working with anyone who isn't grinding. I'm giving all these hours to you, I just don't have the time for it. It's like you're inventing an algorithm in your head, the more you program the more you're able to figure out the structure of the system to get the result you want. It's not that hard for programmers like us, to think of algorithm style solutions. Ok so, I need to kick them out. How do I do this?

I have done a lot of internships, and remember monday updates and daily standups. So I got everyone to do frequent check-ups. Publicly saying what they did to the full discord server, that keeps them accountable. The second one was the hours they were doing, I wanted everyone to do 30 hours. And that's the trust system, a lot of people are trustworthy, I guess some people aren't truthful. That alone isn't enough, initially I didn't have enough requirements to make sure people were doing enough, so people needed a kick in their bum and I adjusted the requirements that you need to pass certain milestones by a specific date just to make sure there aren't too many stragglers behind. I kept on tweaking knobs to figure out a machine that creates Google interview super soldiers.



It's sometimes so difficult to put things into practice and begin your journey when there's so much noise online, in university, and online tech gurus. They give all this advice, and it can get so saturated and we don't know what the source of truth is. What is the antidote to this?

There are many, many pathways to get into your dream position or start your own company. A part of the problem is that people want a quick fix solution, so they go to reddit, they go to Youtube, they go to blog posts and they want to find out what's the secret. And Youtubers know that people are going to click on those videos because people want the "secret", so they give a lot of bite sized information that's good for the youtube algorithm and that's their job, it's to give us information. Their job isn't to guarantee you getting into Google, their job is to give you enough useful information in a way that's easy for people to listen to.

The most important thing to remember is that everyone online, their motivation isn't actually to get you a job, because they have their own career and they just want money out of youtube. The hard pill to swallow is that, what you need to do, it is going to be the hardest thing on the list. There's all these things you need to do, and if one of them is to grind 300 leetcode problems and spend the next 6-8 weekends on grinding that sucks! But, that's what the solution is, and there is no way around it. And you had a more direct question: Given all the noise, how do you find a sense of truth? So, your truth is different to everyone else's truth. It sounds like a bad answer, but I'm coming to a point here. What you need to do and what you need to figure out is: "What is your big impossible life goal that you would love to have in 5-10 years", and don't make any concessions on that goal. What is that? Do you want to work for NASA? Do you want to work for Google? Do you want to start your own company? You know, don't make concessions and think "I might not get there", because this is your lofty goal. And, what you need to do is figure out a plan to get it.

So, I've got a few talking points I want to get to before I go back to this plan. Because the plan is like the end of the advice I want to give. The first talking point, the first point is why you should even have lofty goals in the first place? I want to work for a startup company, I don't want a big Google internship. But you need to ask yourself a question, if Google and a startup company gave you an offer, which one would you choose? And most people would wanna pick the big company, even if they tell themselves "I want the small company". Even if someone genuinely does not want to work for a big company, because some people don't like the monopolistic tendencies of these companies and they don't want to help feed that machine, people might genuinely not want to be at those big companies.

However, there is a problem when we set easier goals for ourselves. So, if you set a goal that is too small, and that goal is, "I want to get myself into this small company", then that might be pretty easy and the amount of work might be small. And human nature is to move very quickly to get to a certain point, and then 80-85% of the way there we slow down because you think we've already got there; "I don't need to work that hard anymore". And if you do this for your small goal, you'll make that small goal but it will take longer because you'll slow down 80% of the way there. But, in comparison, if you set a big crazy thing as your current goal, then what is going to happen is you're going to move very very quickly until you're 80% of the way there and then you'll slow down. Instead of having small goals where you go really really fast, then slow down, then really really fast then slow down; that's where this big goal comes into play. So that's the first thing, why you should aim for the big goal straight away.

And then there is another idea, it's nice of us to say the words NASA, Google, Microsoft and say I am going for that. And this comes more back into the question you're asking, of how do you find the truth in all the noise. And the first thing you need to realise is that is that a goal or is that a dream?. And there are two big differences. If it's a dream, the dream will not come true. It's just a thing that if you dream of, and you do the right thing here, then this thing might happen and you might fall upon that in the future. A goal is something that will come true, and the difference is that you have explicitly made a plan. You can create this plan by getting information online and watching youtube, and all of it is valid because that person HAS gotten into Google so obviously what they said worked for them, etc... You can talk to me, other people in your community who have also done it as well. But what you need to do is take that information and turn it into your own plan.

So take all of this and say you want to get into Google. Ok so what's the halfway plan? Maybe the halfway point is "I am confidently ready to interview for Google and my resume is good enough". All right, you need to know what has to be on the resume and that's when you can use all this information online. You can use RSP, online resources, but even that might be too far away. What's the 1/4 point? Maybe you need to get some start-up experience, maybe you need to do a project with some friends, maybe you need to start grinding leetcode. You should write down the entire plan, from start to Google, and you need to explicitly state points. You can change your end goal, mine has changed many times, but if Google is your goal and you're currently here, as you walk along your timeline, you can check that you're 25% of the way to Google now. Let's say the goal changes and you don't want Google, you're still 25% of the way there and chances are the thing you want, what you've done is already going to help you get into this new thing.

So basically, you can use information online, but YOU need to write your own pathway and you need to plan your own journey and don't just blindly follow what they think is the right thing to do for your career. You need to be explicit about what you want, and then you need to make your own plan for that goal. And therefore, you can use all this information online to help with your goal.

That's really interesting, the concept of the dream vs the goal and their differences

Yea, most people you meet at university, everyone has a dream, but the majority, only like 15-20%, it's actually a goal because they've actually planned for it. Most people don't plan for it therefore it's not going to come true.

That really made me think of Elden Ring. We have this ultimate goal, and there are these little checkpoints that you have to pass through. And even thoughIcangotowardsadifferentboss, I still followed those checkpoints and it did eventually help me to my next boss. Or another boss if I do change my goal.

And it's important, because there are so many things that are valuable in our time, that could be valuable for a career, but if you don't actually solidify this goal that you are going for, what everyone does is a breadth first search. They say yes to this small job over here, and say yes to this thing over here, and "it's all good experience" and therefore "that'll get me to where I want to be". But if you want to reach some really far point, you don't use BFS, you use djisktra's, you use A*. And A* will point you more in the direction as you move and will have more intelligence in the things that you are doing, and in the exploration that you are doing. That's the programming way of thinking; people who don't have a goal or something they're working towards are doing this breadth first search style with their career and their life. Just moving in all directions blindly, without focusing their energy on: this step will get me to that step, which will eventually get me to Google. They haven't talked about the DFS, the depth first search steps of the things they are actually doing.

Special thanks to *Ravi Hammond* for taking the time to attend this interview despite a busy schedule.



Artificial Intelligence V/S Natural Intelligence A Contrasting Comparison

"The development of artificial intelligence will be either the best or the worst thing to happen to humanity." - Stephen Hawking

Overview:

How many times are we going to hear the term "AI" this year? Probably a Zillion, a number with an indefinite number of Zeroes. You might be confused by now – but it is true that there are indefinite number of zeroes in a Zillion and as for the confusions about the future with AI – well that's not going to end anytime this year or later. What's the AI hype really about? Are human beings really going to be replaced? Is AI just another technology? What's the banter? More importantly - how far can this AI hype go and what will it actually deliver to humankind at the end of the Gartner hype cycle? (If you do not know what the Gartner hype cycle is, please read about it out here.)

Let's jot down to our first principles thinking for a second and examine how intelligence is defined until the present age? I say until the present because I personally think in the coming few decades or say at the end of this century – intelligence itself will have an entirely new meaning which most people cannot fathom to understand as of now.

What is AI?

To explain to you in one sentence and save you the previous time – AI is just a system designed to replicate human cognitive function or simply human intelligence. That's it! See you all folks. Now you have cracked AI (quite literally!!). I can even one up myself and explain to you much better with a picture what AI is.



But wait, surely this cannot be just it – else why would there be so much hype around it? Well, it turns out AI cannot just replicate human cognition, but it can outperform it by hundreds and thousands of folds. Sucks to know this isn't it? Well, that's one way of belittling a person's ego – just tell them their mobile phones (with AI cough) are smarter than them (warning does not go well on your bosses or managers apparently).

Let's get back to our scientific analysis, the modern-day AI is powerful, we have come far from considering an abacus as AI to self-driving

cars and all-knowing Large Language Models (LLMs). Right from the start of the digital age roughly around 2002 - there have been efforts to form a complex and mutually constitutive mirroring of the brain, the human mind and the computer (Bates, 2024).

How Smart are our AI tools of today?

Let's take in the state-of-the-art large language, generative AI model today like GPT-4 from OpenAI which is estimated to have around 1.8 trillion parameters and the amount of data that it has been trained on massive 1 petabyte of data which is 1000 terabytes (TB) or a million gigabytes (GB) (Guizeni, n.d.). Most modern-day laptops come around 1 terabyte of data, so the data in GPT 4's mind is equivalent to a thousand laptops of fully stored data.

My personal take on the above is while everyone including experts do evaluate higher number of parameters in an AI model leads to smarter and better models – it is not set in stone that this is true. Several models with smaller capabilities outperform the larger models, especially in more specific tasks or data which these smaller AI models are trained on. Having said this it is a known fact that with higher parameters and better computation available to train the AI models something called as 'Emergent Intelligence' is discovered in the AI models where it correlates reasoning across domains and helps in connecting the dots between events or behaviour between different domains. Emergent Intelligence in an AI model can be described as creation of new and unique game strategies in a game like Chess or AlphaGo by the AI which human beings would have never thought of or heard before. This aspect of AI is not completely understood yet and is often hard to interpret by the best in the business. I would have liked to go further and compare different language models with each other and compare different types – language, vision, multimodal, audio AI models among themselves to see which ones are smarter, parameters, amount of data trained on and their interpretabilities but I would keep that for another article (maybe my next one) and explore more of Natural Intelligence in this one.

A special note to recent developments in simulative AI technologies like– AlphaFold 3.0 and Google Knome. If you didn't know this already AlphaFold can simulate the protein interactions of living cells and find the cure to diseases in a simulative environment – all with no or limited actions required to be performed in the real world. Think of all the medical trial's test mice we would be saving due to this AI tool. There is also Google Deepmind's Graph Networks for Materials Exploration (GNOME) which could predict 2.2 million technologically viable materials all using computing power and algorithms (Google DeepMind. (n.d.)).

Natural Intelligence: The Leap of the Way

"Thou shall not make a machine in the likeness of a human mind." - Frank Herbert, Dune

Microscopic tiny little cells randomly moving across different textures of planet Earth for a chance to survive further – yes this is what Intelligence in its first known form was. Let that sink in!! Base intelligence was not intelligent you would say but humans' beings and all other beings with any form of intelligence. From random cell movements to designing spaceships to travel to different planets – this is what I would like to call as the leap of the way of Natural Intelligence and all the known history of mankind is a small subset in this storyline.

Having said this, what still is Natural Intelligence though? Collecting data from environment and deciphering it into thoughts and information and performing actions based on that is how I would best explain it – in simplified and generic terms. Emotions, motives, desires, personality, skill, foresight, intellect – these would all be outcomes of intelligence based on one's biological structure and consciousness which we will term is as one's beyond. Now we could write an entire article about consciousness but let's leave it to some other day. Principles, laws, science, proofs, algorithms, theorems we can deem this to be the followed pathways of intelligence or pathways leading "to" intelligence – it's like a chicken and egg problem. Does Intelligence? That's something for you to ponder upon. While we can ponder upon many such dilemmas related to natural intelligence, one thing to be of prime importance is that the evolutionary tree of our species is deeply rooted in the development of natural intelligence, once fully formed animals started adapting to their environments, complex nervous systems and brains became essential for survival and further development of intelligent functions. Human cognition, which I can best describe as the understanding of the situation one is in and developing a strategy that's best for one's survival and improved conditions of living is the pinnacle of natural-based intelligent systems.

What makes Natural Intelligence unique?

Natural Intelligence, which even though may seem a tad bit slower than AI – after all, as we humans don't have GPUs attached to our brains. To give you comparative computational reference compared to the latest LLMs, the human brain has about 80 billion neurons and the highest number of parameters or neural networks in an LLM is Google Bard with 405 billion parameters (Zaheer, n.d.). Even though the numbers of LLM seems larger it does not imply that the smartness and architecture of the LLM is much better compared to the human brain. Let's look further into what an individual neuron is in a human or biological brain.



Note. Synaptic Transmission by SimplyPsychology.

The neuron (a nerve cell with its processes) was first accurately described in 1740 by the Swedish scientist and philosopher Emmanuel Swedenborg. A century later, Ehrenberg, Remak, and Purkinje acknowledged the nerve cell as a crucial component of the nervous system (Fodstad, 2001). Neural capacity refers to a neuron's ability to process and transmit information, encompassing multiple factors rather than a single quantifiable value. Key elements influencing neural capacity include the number and strength of synapses, extent of dendritic arborization, firing rate, neurotransmitter diversity, spiking patterns, and plasticity. A neuron with more synapses, stronger connections, and a more extensive dendritic tree can receive and integrate more information. Higher firing rates and diverse spiking patterns allow for faster and more complex information processing. The ability to use various neurotransmitters and adapt through plasticity further enhances a neuron's computational flexibility. (*Brain Basics: The Life and Death of a Neuron, n.d.*). Together, these factors determine a neuron's overall capacity to contribute to complex neural networks and brain functions.

Natural Intelligence does have some quite unique aspects to it which AI cannot really excel at or say replicate as of now which are:

- Emotional Intelligence: Humans and several animals have the ability to recognize, understand and manage their own as well as other emotions. These emotional capabilities is very unique, cannot be quantified to replicate in a computer system and plays a crucial role in decision-making, social interactions and overall well-being leading to prosperity and progress of us biological creatures. Emotions also help us the most with ethical meaning and in deciphering which action is right, and which is not according to moral standards.
- Intuition: The ability to make decisions and understand scenarios even with incomplete information – the likes of gut feelings, déjà vu and the undiscovered mysteries and hidden information that lies in each and everyone's DNA is something that helps us feel a deep connection with this physical realm that we live in. It may be speculated that the DNA holds the memories and evolution gains of our ancestors which explains how we go on to have intuition in the first place. Think of our DNA as some sort of incompletely understood pretrained model perhaps? This intuitive capacity of ours often complements our logical reasoning in problem solving and decision-making like no AI can as of now.
- Multi-Modal Mastery: The ability for humans to correlate information across multiple modes – mainly audio, vision, smell and touch and fill in the missing pieces of information of one mode from the other is way advanced than we actually realise. The fact that we can seamlessly integrate and utilize multiple sensory and cognitive modalities to get information from is a fascinating art from thousands of years of optimization. The world-famous musicians and athletes of the world are living proof that we have extraordinarily evolved in this aspect.
- Consciousness and Awareness: Natural Intelligence brings in a sense of self and autonomy to a biological creature. It contributes to our ability to reflect on our own thoughts, emotions and experiences and find a deep meaning in them. I believe this also makes us such great storytellers and dreamers of a better future.

 Creativity and Innovation: We all know what creativity and innovation are, so I will skip being a bot trying to explain it again, and just compare this aspect between AI and NI. Now, I feel this is the most interesting one and I saved it for the last. I have watched numerous podcasts and read several articles about whether AI can be creative or not.

I mean, what is creativity anyway? Is it merely a bunch of permutations and combinations of mixing and matching the existing knowledge and voilà!! You create something new and novel? I think creativity and innovation is truly more than that and it is best associated with imagination. Hear me about – there is no limit to human imagination. Are you truly limited in imagining something based on the data that we currently have? I don't think so, a spontaneous that unrelated to anything I have heard or seen about can emerge in my head out of nowhere. How is this possible? I do understand that yet, but can it be done? A definite yes. Now let me give you an example – if you ask a thousand different scientists on how to create a new space travel system which they have not heard in any sci-fi movie or TV show or read in any research paper about the chances of getting a truly unique answer is high all thanks to our imaginations.

Now I am sure this feat cannot be achieved by any AI yet as they all are reliant on the data, they have been trained upon, but I definitely do not say this cannot be achieved in the future thanks to the rapid advancements of AI. Maybe that's what they call an ASI – Artificial Super Intelligence? Let's wait and watch – it might take a few years though for this to happen.

Future Implications: Fusion of AI and NI?

Brace yourselves and behold the future!! The most intimidating thing in the universe is the future – for everyone thinks the know what's coming yet history has showed us that we are quite often wrong! The most experienced economists and futurists have often got their predictions incorrect more often than not.



Note. Eternity by Marvel Cinematic Universe Wiki. (Fandom, n.d.)

As AI continues to advance potential implications could be the following (notice that I use could instead of would here!):

- Job Displacements: The first and foremost that we are experiencing already. Due to the increasing capabilities of AI in generative speeds, automations – simple tasks can be done in a click a button or a ChatGPT prompt rendering several low-skill jobs be cut off.
- Data Security: No data or creative art is free from AI as sooner or later every new and novel piece of art or science will find its way into the pre-training data models only for the AI algorithms to gobble it up and digest it (quite literally). That being said, the laws and regulations on whether Personally Identifiable Information (PII) will still hold the same standards is one to watch out for. Case in point as well are the rise of technologies of quantum computers and post quantum cryptography which will likely be the best defences for data security in the future.
- Human-AI collaboration: Now we all know we can access AI functionalities through our smartphones and laptops can carry it along with us wherever we go. What seems looming in the foray is whether microchips and brain implants will become part of the human body to go real 'Android mode', again this is not the reality in some dystopian universe as we already had microchips implants in human brain thanks (or not) to Neural link.

I wish we had a third type of Intelligence, which could analyse the pathways of evolution of NI and AI and predict how NI and AI will be collaborated in the future. If anyone knows of such a system, please do give us a ring!





Note. The new concept universe Solarpunk by Guzel, Z.E. (Medium,2024).

Conclusion

I, the author's personal take is that - AI is independent of the Gartner hype cycle, AI will be used as a fundamental base for other technologies to develop and improve upon. We need to dig deeper in history how changes in fundamental levels of knowledge acquisition have reshaped society and what kind of impacts it had on society, but I assume we would just get clues from it on how to deal with AI and definitely not the complete solution. Also, there can really be no end to Intelligence in this universe and beyond, whether that is natural or artificial does it really matter? More importantly, what can we do with the it?

Article by Jeevan Share your thoughts with us at wired_academics@gmail.com.

References:

- Bates, D. W. (2024, April 30). <u>Read an excerpt from An Artificial</u> <u>History of Natural Intelligence. The University of Chicago Press</u> <u>Blog.</u>
- Brain Basics: the life and death of a neuron. (n.d.). National Institute of Neurological Disorders and Stroke.
- Evans, O. G. (2024, February 2). <u>What happens at the synapse?</u> <u>Simply Psychology.</u>
- Fodstad, H. (2001). <u>The neuron theory. Stereotactic and</u> <u>Functional Neurosurgery</u>, 77(1-4), 20-24.
- Gartner Hype Cycle Research Methodology | Gartner. (n.d.). Gartner.
- Google DeepMind. (n.d.). <u>Google DeepMind: Building AI to benefit</u> <u>humanity.</u>
- Guizeni, S. (n.d.). Unveiling the immense scale of GPT-4: A deep dive into the model's size and capabilities.
- Güzel, Z.E. (n.d.). <u>The new concept universe: Solarpunk [Image]</u>. Medium.
- Marvel Cinematic Universe Wiki. (2024, October 27). <u>Eternity</u> <u>Infobox [Image]</u>.Wikia.
- r/ProgrammerHumor. (n.d.). Artificial intelligence versus natural [Image]. Reddit.
- Zaheer, N. (n.d.). <u>8 best large language models for 2025</u>. Educative.





AlphaGeometry: a theorem prover for Euclidean geometry

Overview:

In this article, we explore AlphaGeometry, an advanced theorem prover designed for Euclidean plane geometry. This system significantly reduces the need for human intervention by generating synthetic theorems and proofs, which are then used to train the model. At its core, AlphaGeometry is a neuro-symbolic system, combining the strengths of language models with a symbolic deduction engine. Impressively, AlphaGeometry has successfully solved 25 out of the 30 most recent olympiad-level geometry problems, producing human-readable proofs. This article provides a high-level overview of what AlphaGeometry is and how it operates.

What are Transformers?

Before diving into AlphaGeometry, let's first talk about transformers. Developed in 2017, transformers are now what power of many advanced language models, including GPT-4, which I am sure many of you used. So, how do transformers work? Imagine you're trying to solve a puzzle where each piece is a word in a sentence. A transformer helps by looking at the entire sentence (or puzzle) all at once instead of one word at a time. This allows the transformer to understand the context of each word, making it better at tasks like translating languages or generating text. A transformer is made up of two main parts: an encoder and a decoder. The encoder takes in the input (like a sentence) and turns it into something the model can understand. The decoder then takes this and turns it back into a sentence, but in the way the model wants to output it (like translating from English to French). This ability to handle context and generate meaningful outputs is why transformers are so powerful.





isual representations of the synthetic data generated by AlphaGeametry

Note: Figure 1 (left) illustrates the Transformer architecture, showcasing its key components and data flow. Figure 2 (above) presents a visual representation of synthetic data generated by Alpha Geometry.

Generating Synthetic Data

One of the cool things about AlphaGeometry is that it doesn't need humans to show it how to solve problems. Instead, it learns by practicing on synthetic data. Researchers generated one billion random geometric diagrams, each containing points, lines, and shapes, and the relationship between the points and lines in each diagram were derived.

These synthetic diagrams cover a wide range of possible geometric scenarios. By studying these diagrams, AlphaGeometry learned the rules of geometry without needing examples from Figure 1 Figure 2 humans. This massive dataset is what allows AlphaGeometry to tackle new problems it has never seen before.

Figure 2 in the article would show you what some of these synthetic diagrams look like, giving you a glimpse into the training process.

How does AlphaGeometry Work?

AlphaGeometry's secret sauce lies in combining two powerful tools: a language model and a symbolic deduction engine. Let's break that down.

1. Language Model: This is where transformers come into play. The language model in AlphaGeometry is great at spotting patterns and relationships in data. Think of it as having a really good intuition for what might be important when trying to prove a theorem. However, while it's good at suggesting ideas, it isn't always able to explain why those ideas are correct.

 Symbolic Deduction Engine: This part is like a strict math teacher. It follows precise rules to prove theorems, making sure everything adds up logically. Unlike the language model, the deduction engine doesn't guess, but rather it follows clear, step-by-step reasoning.

AlphaGeometry put these together by getting the language model to take the infinite number of constructs that can be added and predict which new constructs would be useful to add. The symbolic deduction engine then takes these new contracts and then tries to solve the problem. If the problem is not solved, then the language model is consulted again. This process is repeated, until the problem is solved. Hence the language model guide the symbolic deduction engine through infinite branching points in challenging problems.

This combination of intuitive guessing (from the language model) and rigorous checking (from the symbolic deduction engine) enabling AlphaGeometry to solve very difficult problems.



Note: Figure 3 (above) shows an overview of AlphaGeometry solving a geometry problem.

Conclusion

AlphaGeometry is a remarkable tool in the world of automated theorem proving. By blending advanced neural networks like transformers with traditional logicbased methods, it can solve some of the toughest geometry problems out there. For computer science students, AlphaGeometry offers a glimpse into the future of Al in mathematics where machines can not only solve problems but also explain their solutions in a way that we can all understand.

> Article by Nish Share your thoughts with us at wired_academics@gmail.com.

> > 23

	u na seconda da second
	<u> </u>
▖▖▕▖/ <u>シ</u> ╢▆ᢞᠾᠫ <u>ᢒᢄᢟᡈ</u> ᡜᡦᡐᡄ᠊ᡧᢧ <mark>ᢧᢟᠼᡷᠥᡘ᠅</mark>	
- Grattagor (regerad	, kolo produkter i se
KYZQ IP PZZ(LEJILYK	withikhiggrad digayji
de la companya de la	
en kärste Deussen Staffasträhtt	en a prime de la constante de la const
	ongevæder i
- Kikywengeryieiii	
<u>— Elympic Strated de Carso</u> u	
nlling 7a	