/mlst

Moving Beyond Surface Statistics (Apple researcher) [Iman Mirzadeh]

March 23, 2026



Machine Learning Street Talk

Podcast Transcript

Revision: 4cc3ecb6

Topic headings include retention indicators: ● above average, ● average, ● below average (vs. similar YouTube videos)

Interested in licensing ReScript for your podcast? Get in touch with Tim Scarfe.

Contents

Overview 2

Intelligence vs Achievement in AI 4

AlphaZero and Abstract Understanding in Chess 5

Language Models as Distribution Learners 7

The State of AI Research Methodology 8

Interpolation vs True Reasoning in LLMs 11

Measuring Intelligence: From Chollet to the Iman Moon Test 12

Agency, Active Learning, and World Models 15

Scaling Laws and the Connectionism-Symbolism Debate 19

GSM-Symbolic: Exposing LLM Reasoning Fragility 23

References 29

Overview

Iman Mirzadeh is a machine learning research engineer at Apple and the lead author of the GSM-Symbolic paper, which exposed deep fragility in how large language models handle mathematical reasoning. In this conversation, he draws a sharp line between intelligence and achievement – between what a system can score on a benchmark and what it actually understands.

The discussion starts with chess. Mirzadeh explains how grandmasters don't use engines to memorize moves; they use them to develop theory. AlphaZero discovered unprecedented strategies, but that knowledge stays trapped in the game. Humans, by contrast, extract abstract principles like 'control the center' and transfer them to entirely different domains. That capacity for abstraction is what he thinks current AI architecturally lacks.

His critique of LLMs is structural. These systems are trained to minimize cross-entropy loss over a distribution, and by construction they cannot reason beyond what that distribution contains. Prompting conditions the learned distributions but doesn't create new understanding. Change the surface form of a problem – swap names, add irrelevant clauses – and performance varies wildly, even on grade-school math.

That's the core finding of GSM-Symbolic. By generating templated variants of math word problems, Mirzadeh's team showed that even frontier models exhibit large performance variance from changes that should be semantically irrelevant. The GSM-NoOp variant, which adds sentences carrying no arithmetic operation, causes further drops. The implication: what looks like reasoning is closer to sophisticated pattern matching across memorized distributions.

Mirzadeh proposes that intelligence should be measured by the slope of a system's scaling – how fast it can learn novel things – rather than its current benchmark position. He offers a striking thought experiment: give MMLU to a caveman and performance would be near zero, yet no one would claim a language model is more intelligent. He draws on Chollet's skill-acquisition-efficiency framework and Gilles Gignac's formal definition emphasizing novelty and time-bounded capability.

The conversation also covers the connectionism-symbolism divide. Mirzadeh takes the position that symbols could in principle emerge within neural computation – middle activations might serve as formal symbols – but that current training regimes don't produce the kind of abstract world models needed for genuine reasoning. He sees active engagement and agency (drawing on Stanislas Dehaene's work) as necessary conditions for learning, not just nice-to-haves.

Throughout, Mirzadeh is careful to say he doesn't have all the answers. He's call-

ing for the field to slow down and build theoretical frameworks before optimizing solutions. The airplane-to-the-moon analogy captures his position: you can make the plane faster and lighter, but that won't get you to the moon. Maybe we need a different vessel entirely.

Intelligence vs Achievement in AI

Iman Mirzadeh 00:00:00

Think if someone wants to take 1 important message from this talk, I would be understanding the difference between intelligence and achievement. The field is currently focused very heavily on the achievement and, the numbers and accuracies instead of trying to understand what does an intelligent system mean, what does it mean for a system to understand, to reason instead of tying it to a certain number for a certain benchmark?

Tim Scarfe 00:00:28

Iman, I think you would agree that we need to have better abstract world models. Right? We need to have better representations. How's that

Iman Mirzadeh 00:00:35

gonna work? To me, it looks nearly impossible to build an intelligent system that operates without an abstract model of the environment and the world and knowledge. But right now, there are many questions that needs to be answered before that. 1 of the issues that I have with the literature right now and also including me is that we don't even have basic answers to these questions. So after alpha 0 came, what happened is that the chess event became more popular and the the quality of chess improved. Not because people used chess engine and just memorize move, it's because they try to understand what alpha 0 is doing, other chess engines are doing. So grandmasters actually use these tools a lot, but they don't go and memorize the moves by chess engines. What they do is that they develop theory. So they work on novel openings. They work on novel moves at certain position, novel strategies. They learn from these tools, not in a way that just memorizing. They understand. They create a knowledge, new knowledge, new theory. In image and computer vision, we had these benchmarks like ImageNet and all those benchmarks, and we saturated them. And we thought, okay. The vision is solved, but now we see that self driving car is not becoming a thing right now because it's very difficult. Like, in real world, there isn't a specific frozen cut off reality that with fixed number of example. You have to build an agent that understands and reason. So that's why I think it's important to focus on, not the exact number.[1]

[1] Mastering Chess and Shogi by Self-Play with a General Reinforcement Learning Algorithm — paper

Tim Scarfe 00:02:18

MLST is sponsored by Tufa AI Labs. Now, are the DeepSeq based in Switzerland. They have an amazing team. You've seen many of the folks on the team. They acquired Minds AI, of course. They did a lot of great work on ARC. They're now working on o 1 models and reasoning and thinking and test time computation. The reason you wanna work for them is you get loads of autonomy, you get visibility, you can publish your research. And also, they are hiring as well as ML engineers. They're hiring a chief scientist. They really, really want to find the best possible person for this role, and they're prepared to pay top dollar as as a joining bonus. So if you're interested in working for them as an ML engineer or their chief scientist, get in touch with Benjamin Crouzier. Go to tufalabs.ai and see what happens. You know, we all accept there are there are limitations in reasoning in in LLM systems, and some people say, oh, it's just because there's a missing feature, a missing tweak. We just need to tweak the transformers architecture, and then they can do copying and counting and reasoning, all those stuff. Other people say, all we need to do is let them access tools. Mhmm. Right? So for example, like I wanna have a system that plays chess. It can just use a chess computer. What could possibly go wrong with that?

AlphaZero and Abstract Understanding in Chess

Iman Mirzadeh 00:03:28

So yeah, that's 1 of the most common arguments I hear and debate with many friends and colleagues and people, even at NeurIPS. And I don't personally have any issues with using tools. Humans also use tools. Everyone uses tools. So but going back to our previous part of discussion, it's not about solving a task. It's not about achieving certain accuracy on a task. Then it's about understanding. It's about creating novel knowledge, creating novel goals, achieving those goals. So there are 2 arguments I have against the tool use. So 1 is that, first of all, it's not just 1 tool, Right? So there are like, for chess, yeah, if your only goal in the world is to do well on chess and beat some humans, yes, you can look at, I don't know, a chess engine as a tool and just gives move give moves to it and get moves and play chess and win. So it's it won't be just 1 tool for the for your whole life. It will be many tools. Right? So if you look at some reasoning tasks, right, like solving a math, logical problem, planning, so it it involves many steps, many many states that you need to navigate and learn. And even with tools, you won't be able to use won't necessarily will use only 1 tool. You will need many tools. So it will be like, I don't know, at some point, you will need to use 5 tools, 10 tools, and then the tools may become compli-

cated. They might not necessarily be like you give input and you get output. It will be complex. So essentially, it will reduce to the same problem, right? What if I need to plan for 10 different actions if you look? And so at that point, you you still face the same problem. Right? So that's 1 argument. And for the chess, also, there is this argument that if you use tool, it's fine to use tool definitely, but you don't necessarily be able to know if the system that uses external tool understands that task. So looking at the chess engine, you could play you could use a chess engine and play and win, but you won't necessarily understand why did you make those moves, why why there is this move in that position, what what constitutes a good position, bad position, planning. So actually, because I'm, like, a huge fan of chess, I I watch chess competitions, and so I follow what's going on in the chess world. What happened is that after alpha 0 came, so humans instead of like everyone thought, like, the chess as a sport and the competition will be kind of will lose its importance because there is the system that wins any grandmaster in the world. But what happened is that the chess event became more popular, and the quality of chess improved, not because people used chess engine and just memorized moves, it's because they try to understand what alpha 0 is doing, other chess engines are doing. So actually, they wrote a book on understanding the moves of alpha 0. Right? So the famously alpha 0 has this move that sometimes it tries to push from the left or right corner, and this moves have not been, like, seen before. Right? But you could say that I don't care. This is a move that goes like, gives me advantage and wins, or you could understand why this move this kind of moves improve chess. Right? So grandmasters actually use these tools a lot. Right, but they don't go and memorize the moves by chess engines. What they do is that they develop theory. So they work on novel openings, they work on novel moves at certain position, novel strategies. They learn from these tools, but they are not in a way that just memorizing, they understand, they create a knowledge, new knowledge, new theory. And I think this distinction of like how humans use chess engine versus how people think we should use chess engine is very important. So if someone claims that chess engine like you can use chess engine for your LLM as an external tool, I'm fine with that if you wanna develop that system, but my metric for measuring how good that system is is that if you start if you let that in system or LLM or any system use that chess engine, after a while it comes up with a theory. So 1 example would be that human like in chess, there is this principle that in the opening phase you should have a control on the center because you have control over more squares, it allows you to like in the next moves, it allows you to have more squares and it prevents your opponent to develop its pieces. Right? So development of pieces controlling the center are the principle of the chess. So if you start with any system and a chess engine and after a while, it comes up with why it is important to control the center because the openings like you could

train a model on all the games and it will learn as a distribution and statistics that the first moves are for white e 4, d 4 or c 4, right, like the first the the center pawns. So the reason behind that is that they they wanna control center, right? If it comes with that new knowledge that understands why it matters using a tool, it's fine. But if it's just, you know, using a chess engine and gives moves and back and doesn't have any understanding on what's going on, then all you did is that you got, like, high rating on chess without really knowing what's going on.[2]

Language Models as Distribution Learners

Tim Scarfe 00:10:10

How puritanical should we be here? So, you know, LLMs, they act as if they are doing reasoning in many, many circumstances, but they're prompt dependent. You give them the wrong prompt, you put distractors in there, it brittleizes it, it it it doesn't work. So how do we interpret that? Do we do we do we say it's not reasoning because of the brittleness?

Iman Mirzadeh 00:10:30

The short answer is yes. So the way I look at the prompting and the way and the reason I also don't necessarily think that, like, exploring different prompting helps in terms of really understanding, like, improving the systems is that I look at the prompting is that I look at LLMs as learning as a system that knows a lot, lot, lot of distributions. Right? And then the prompting to me looks like conditioning those distribution, nudging those distribution to, like, maybe specifically steer the model towards some direction. And so you could use prompt to do anything you want almost, but that doesn't mean that the system is understanding what's going on behind the knowledge. So my problem with the classical way of what the way we train the system is that we look at everything as a distribution. And when you learn a distribution by by construction, our loss function is that you learn a distribution that minimal and minimize the distance between your distribution and the sum distribution that you infer from data. Right? And once you learn that this week, that's all the model cares. So once you learn that distribution, you just stay in that distribution. And that the problem is there are many problems with that. The first 1 is that by definition, by construction, these systems won't be able to understand what is beyond that distribution, right? By construction, we train the models to minimize the loss and stay, like, within a boundary of a distribu-

[2] Game Changer: AlphaZero's Groundbreaking Chess Strategies — book

tion. And that's what leads to all the problems. Right? You change something, system breaks. You want to measure how the models perform on different distributions, they can't do that. And all these problems, because that's how we trained our model. How do you expect the model to stay in a box? And then you ask it, what happens outside the box? And you constructed that system this way. So I think that's 1 of the issues. And another issue with the way we train, I think, our models. I don't have, by the way, answers to how to fix this right now. But another issue is that when the way we train with, for example, cross entropy loss, right? So let's say you are like, you teach you are teaching models to do arithmetics. Right? What we do is like 2 plus 2 equals. There are like, assuming there are like 2 is a token plus is a token. So these are the tokens. And then in the text, there is 4. Right? So all the model has to do is learn after 2 plus 2 equal, it should output 4 and some context maybe. So it doesn't matter for for the system that this 4 comes from understanding the whole natural numbers, the the the line of numbers, addition, can we do this or not, or it's just memorization or maybe some other thing, right? So there is nothing in our loss function saying like, it's important that you understand numbers and natural numbers as a concept, which later will be grow to, I don't know, rational numbers or like real numbers and all this concept. All you have to do, all I care is that after 2 plus 2, you should give me 4. Right? And because of that, we don't know if the model can come up with that understanding and building a world model or kind of concepts of what's going on. But yeah, I think these are the main issues with the like all those these are the consequences of, I would say, all the the all the things we are doing to train the systems.[3]

The State of AI Research Methodology

Tim Scarfe 00:14:47

So you said to me that you're overwhelmed by the lack of progress in, you know, the sphere of intelligence in AI research at the moment. Tell me more.

Iman Mirzadeh 00:14:57

Oh, okay. Yeah. I mean, there's like there are 2 sides to this story. 1 is that, obviously, I think we are living in a very important time and very exciting moment, and there are lots of progress being made. Lots of people are investing in AI right now. There's lots of research being done. So from that side, it's very exciting. On the other hand, I think, like, in my opinion, the way we are doing research in the field is not necessarily optimal. And the way we designed our metrics and the

[3] NLP Course: Language Modeling — reference

way the system works in general is not necessarily optimal. So for example, right now, the the current way of doing research, most of the research is that people believe that research is an an incremental pro process, and I don't disagree with that. You you will always build on top of work of other people. But my my issue with how the research is being conducted right now is that we don't by reading new papers and learning all the new papers, we still compare to, I don't know, 2 years ago, 3 years ago, 4 years ago, we still don't have a better understanding of how these systems work. Right? We do not know that, like, whether we are moving towards the right direction or not. So for example, like, most of the papers these days have a hypothesis is that, like, if you do this, then this will happen. And then the Iran experiment showed that, like, yeah, if you do this, then this happens. Then there will be another paper saying that, if you do this, then this happens. And then there will be other papers saying that all these papers had this assumption. And then if you have a different assumption, this will happen. And then still after, like, reading, I don't know, 10, 20 papers, you still don't have a coherent picture of, like, how these systems work. Like, what is the general hypothesis? So I think what what what is, I think, good about many other fields like physics is that you have a hypothesis saying, like, this is my model of, I don't know, a transformer. Right? This is how I think a transformer works. And then after that, you say, okay. If my hypothesis is true, then there will be these consequences. Right? And then I'm gonna do a an experiment and measure that. But this is now not how things work. The the way people approach this problem is that they they look for solutions with before understanding the problem. So I think that's 1 of the issues that I hope, like, at least people spend more time on, like, understanding things rather than focusing on the solution and what we can do. I I I am always like I think it's more important to understand what is what what is happening right now, what are the problems, what are the pros and cons of these systems, how do the how do these systems work. Right? And then once you have a, like, at least hypothesis or mental model, or, like, how these systems work, then you can build on top of that. But otherwise, you will maybe run into like papers or like research that think like instead of trying to understand how like prompting. Like, you could say my model of prompting is that it will nudge the distributions. It preconditions the model. And then if there are a few distribution and the model is interpolating on that, then this will happen. Right? And then you could run a few, like, test this hypothesis. But, like, what happens is that someone explores that what happens if you tip a language model. And even if it, like, increase your accuracy by 5%, you still don't know what's going on. Right? Like, you you after reading that work that increased the performance of the model, like, a few percentage, like, all those work, you you still don't have a better understanding of the models. And then that will be makes it very difficult to improve those systems. Right? And then there are other, I think, aspects like I think, like, from the theory perspective, I'm not an expert in theory at all, but

I think maybe our theory is also too rigorous, I think, for the moment because the problem is so hard, and the the current theory lacks far behind the state of the the practitioner. And then they wanna stay rigorous, very rigorous, and, like, develop it, but they it's very hard if the theory doesn't, like, become more relaxed, so it can at least be be like, move have a faster progress. So I think those are the those are the main issues. And then from the other aspects, think, like, it's given the amount of interest and investment and in the AI literature, then there are many people working on the field, which is really, really great. But there will be other challenges, like the reviewing process, the peer reviewing process is really difficult and noisy. I don't have a, unfortunately, solution for that to propose here, but I think we can at least think about these problems. How can we improve those processes? And, yeah, I think these these are the main, I think, challenges of the AR literature and the AR research community right now in terms of research and how the research is being conducted.[4]

Tim Scarfe 00:20:58

How can we bridge symbolic AI and connectionism?

Iman Mirzadeh 00:21:03

Yeah. That's a I think, I mean, we there are a couple of ways to think about it. Like, I think you could have a symbolic model. I'm not right now, I'm not saying that, like, this model, like, symbolic model versus non symbolic models or deep learning is good, bad, or, like, I I'm thinking a lot more, like, fundamental ways of, like, given any systems, is there a way to understand if the system is understanding a concept, is able to reason? But in general, I think it is fine to start and, like, integrate these 2 as they're like, maybe we start with a separate module. And then then maybe eventually, like, by marrying these 2 fields, it will be become more integrated and improve over time. But it's very important to, like, not look at these things as 2 separate system. Otherwise, it will lead to problems of, you know, using an external tool and all those things. So there should be some notions of understanding and, like, the model itself should be able to have a word model, should be able to understand if I need to update this or not. So 1 1 other thought experiments that I have is that like, imagine right now that we are talking, I tell you the area of square is p r to the half, like the power of half. I don't know. Something like that. P time pi times 2 r. Right? So you wouldn't accept my argument. Right? Like, you could prove mathematically that this is not correct. But if I tell this the area of the area of square is pi r to the power of 3 to an LLM because there is no module in the model system that's saying, like, is this making sense? What is what do I think about the area of this circle? It should

[4] GSM-Symbolic: Understanding the Limitations of Mathematical Reasoning in Large Language Models — paper

ideally won't be it shouldn't accept it, but we don't have that system. So I think that's 1 of the things that I think 1 of the examples of why we need another component, why we are missing at least 1 component, that after a while, learns comes up with its own belief system and, like, knowledge representation. And if when something comes, it may update its belief system or may disagree. I wouldn't agree that, I don't know, the area of circle is something else because I can prove it to you right now. That that's it. If you have another proof, I will read it, but I will doubt that that proof is correct. Right? So that's very important, but there are other belief systems that I'm open to change. Like, the best the best place to eat in Vancouver, I'm open to change it at. Try your your suggestion. So I think it's important that the the system should to not be separated and, like, be looked at the 2 separate component. It should be a, like, integrated loop.[5]

Interpolation vs True Reasoning in LLMs

Tim Scarfe 00:24:24

Yeah. And I think it it is interesting that you say that humans don't always reason, but we're capable of doing some privileged form of cognition that we might call reasoning. And also, you could argue that the LLMs have a spectrum of modes. So sometimes they're doing surface statistics and maybe in certain circumstances they're doing at least some approximate version of reasoning. So there seems to be a spectrum. Tell me about that.

Iman Mirzadeh 00:24:49

It's Yeah. I think if you look back, obviously, we can admit that kind of what these systems today are capable of kind of surprised the field and everyone, I think. But I think we kind of sometimes get read too much into it and what these systems are capable of doing. And the fact that, like, there there some claims say that the models can reason and, like, there's a spectrum, I'm not sure about that, actually, that these models can system. What they are good at, I think, is doing some sort of interpolation. So it's not that they are just memorizing things. They can't do anything beyond training data. They learn many things from different places, and they learn different distributions. And they do interpolate between these things that they've learned. So I think that's what gives, I think, the illusion of what the these systems are capable of reasoning. And if the domain is very limited and closed, for example, a specific domain where the overall space is closed, the interpolation of the model seems enough to assume that these models are capable of performing in that domain.

[5] Connectionism and Cognitive Architecture: A Critical Analysis — paper

But I think in general, there is a very huge distinction between the achievement and capability of a system and the intelligence and other like, entering the capabilities of the system. So these 2 are different. But currently, what we do is that we mix those 2 together. So for us, right now, it's like if a model performed good at a coding benchmark, it means that it has reasoning. If it does good on math benchmark, it means it's doing, like, some sort of reasoning. And when we mix this together, it's it makes arguments and, like, understanding these systems very difficult. So I think it's important to discuss the distinction between intelligence and achievement. So intelligence, by default, it means like by definition, it means about the capability of system and how it can grow and at some point eventually becomes capable, not necessarily what it is, how capable it is right now, right? So achievement is about measuring what the system is doing on a specific domain or task or benchmark. So these 2 are different. So the if a if a system is intelligent, eventually, this it will be able to do well on some sort of benchmark and but necessary, the other way around is not correct. So if the model is doing well on a benchmark, it doesn't mean necessarily it is intelligent. But because it is hard to measure right to the intelligence in general, this is an open question. We we set a benchmark and say an intelligence system should do good on this benchmark. And then when a model or system gets good, we say this is doing reasoning. So these are different. I think intelligence is about how a system in the long run is able to perform well. So there was a good, very interesting picture in Ilias talk at NeurIPS this year. And and that figure

Measuring Intelligence: From Chollet to the Iman Moon Test

Iman Mirzadeh 00:29:00

showed, like, the there was a regression line between the body mass and brain mass, and that figure showed, like, how the different species scale in that sense. So my take from that picture was that intelligence is not about where you are in that scale right now. It's about how, like the slope of that figure. So if you are thinking about the scaling laws, intelligence is the slope of scaling, not what point you are at.[6]

[6] On the Measure of Intelligence — paper

Tim Scarfe 00:29:35

Yes. I I I love all of this. I mean, you probably know I'm a huge fan of Francois Cholet, And he's always at pains to say that intelligence is not skill. It's adaptation to novelty. It's skill acquisition efficiency. So that there's something about the macro adaptability, which is important. And actually, when we were discussing this before, you came up with your own test for intelligence, which I'm gonna call the iman moon test, which is, you know, basically, we start from cavemen and we land on the moon and and and sort of how quickly can we do that?

Iman Mirzadeh 00:30:11

Yes. So, yeah, that that's that came from a thought hypothesis. Right? So just to, again, reiterate the distinction between intelligence and achievement. Right? So imagine a caveman. So if you go back in time and give MMLU or GSM 8 k to a caveman, I would argue that the performance would be near 0. And current models get near, like, 80%, 90% accuracy on these benchmarks. Right? And do we really believe the current systems that we have are more intelligent than a caveman? So intelligence is about the system that a human possess, not how well it's doing on a benchmark. Right? Even if you go even further, like some great figures in history and science, like Aristotle. Right? If you give MLU to Aristotle, I don't think it will perform nearly as well as current LLMs. But do we really believe those LLMs that we have are more intelligent? Because intelligence is not about how much you achieve on a certain task. It's about how well that system, that human can, if it spends time, learn something and grow.

Tim Scarfe 00:31:32

So how could we measure this? Is it, you know, like, had this formalism for for measuring intelligence, but it it wasn't computable. And benchmarks, they just get saturated, you know, that they they get good hearted. How can we build a new suite of benchmark?

Iman Mirzadeh 00:31:48

Yeah. That that's I think I don't have a concrete answer to that. I'm thinking about it a lot, but I don't think right now I do have an answer to that, but we can maybe not we can at first start with something that is not necessarily objective, but maybe we can start by asking what kind of systems do we want? What does intelligence means to us? What does achievement mean to us? And then if we maybe start with a kind of axiomatic definition and characteristics, then we can at first, it won't be objective, and I know it will be not concrete and open ended. But if you start thinking about these important problems, we may come up with some way of measuring it. So it is, in general, measuring intelligence is very difficult. It's even for humans, it's not trivial to measure intelligence of humans. But I think we can have some define some desired properties that we want from a system. So those desired properties might mean something like the system being able to perform well on novel tasks. And then definition of novel itself is not easy, but I think we can start from something here and then build on top of that and think more on that. So because there was a nice paper by Gilles Gignac et al, and it's called on definition of intelligence. I think that was a really good read. It's not in, like, circulated very well in the machine learning community because they have background in psychology and cognitive science, but it's really, really nice paper. It defines very formally what is the definition of intelligence, what is the definition of artificial intelligence, what are kind of forms of intelligence we are looking for. And we can start from there and think more on those problems, and maybe we come up with something better. So, yeah, that's what I'm thinking on these days. But in general, I think the direction should be towards measuring novel tasks. So for example, if you are training a model to do coding, at least what you could do is that, for me, a better measure would be how fast that system can learn a new programming language from scratch, right? So a system like NLLM could do very well on Python or all the languages it's been trained on. If I create a new programming language and ask it to write a new language, a new program in that language, how fast it can learn that language and how good it can write programs in that language compared to a human that never seen a language before. Because I think 1 other thing that we can do is that compare how humans learn and reason and compare it and contrast it with how these machines are learning and reasoning. So yeah, I think we can think towards that direction, but I don't have a concrete[7][8]

[7] On definition of intelligence — paper
[8] Defining Intelligence — paper

Tim Scarfe 00:35:26

Yeah. It's very slippery. There's a great paper by Pei Wang on defining intelligence, and and he said that it's very anthropocentric. And,

Agency, Active Learning, and World Models

Tim Scarfe 00:35:35

you know, in different fields, people use different techniques. So, you know, we we could have basically a copy of a human brain. And that's the most anthropocentric. That's not particularly useful, is it? Or we could have notions of behavior or more abstract capabilities, so being able to write Python programs or functions. So having things that do planning and reasoning and all these abstract cognitive functions. Or maybe even having things that have certain principles, like, you know, emergence or certain characteristics or something like that. Very, very slippery. But just before we move off that, you cited this paper from psychology. I'm very interested to know what what their definition was. For pay, his definition is very much about adaptivity. That that seems to be 1 of the core things for him.

Iman Mirzadeh 00:36:19

So, yeah, actually, that paper discusses lots of other works on studying and understanding and defining intelligence. I highly recommend that paper. So yeah, the focus of that paper, the main emphasis is on novelty. So the the definition of intelligence in that paper is that the maximum capability and capacity of a system to achieve a novel goal given some time, right? Obviously, you can define many different programs and systems that's given infinite time, eventually, they could reach to a goal. But the the time is also important. That's why I also mentioned the slope of scaling, why and why it matters.

Tim Scarfe 00:37:14

Do you think agency and autonomy comes into intelligence? So, you know, basically, the ability to set your own goals, or do you think we should think of intelligence, you know, purely in the frame of this is a thing, it has a goal, and can it achieve it?

Iman Mirzadeh 00:37:26

No. I think, actually, agency and the system having some concept of what it wants is very important. So I was reading a book recently. It's called How We

Learn, Why Humans Learn Better Than Machines for Now by Stanislas Dehaene. And in that book, he discusses lots of great topics and how humans learn, how machines learn. And it comes from neuroscience background. So lots of new insights at least for me. And 1 of the things that seemed really interesting to me was that that 1 of the pillars of learning for humans is active engagement. And essentially, you some humans like have very it's very hard for humans to just observe and don't get involved and just watch something to learn compared to having them set their own goals, being able to actively engage in the environment, explore, maybe later explore it, and all these things. So without that, that's actually, according to that book, 1 of the necessary conditions. It's not something that is good to have. It's something that you must have. So I think that's 1 of the important topics. And I think for me, the implication was that if you want to build a really intelligent system, supervised learning won't be enough. So you need an agent to ask for what should I do next and be able to ask questions. It's not like you just observe and see, okay. I learned this. I learned that. It should it should be something like, now I learned this. What do I want to learn? Maybe I haven't understood this well enough. I I need to explore more and like.[9]

Tim Scarfe 00:39:26

Let let let let's explore this. This this is fascinating. So controlling the center is is an abstract category. Right? It's it's an analogy. And Douglas Hofstadter said that, you know, a concept is a bag of analogies, essentially. So there's moving to the center is something that we all understand at some abstract level. But what does that actually mean? I think many concepts in chess are quite fuzzy. Right? So what that actually means in many, many different situations is is slightly different. So the question is, what's the difference between alpha 0? It discovered this move 37. And if you play it millions of times, this move 37 in its behavior space emerges as a mode. So you could argue that it is doing reasoning, but it's doing an emergentist form of reasoning. So rather than having some notion of in the center, its behavioral profile acts as if it has a notion of in the center. And how is that so different from the way our brains work?[10]

Iman Mirzadeh 00:40:26

Yeah, I think so again, like, that if so there's this thing, like alpha 0, alpha go, and all these systems are able to understand the environment they're put in. Right? So they they like, they explore the environment, they play a lot, and they are able to eventually explore all the like, most of the kind of positions in Go or chess or many of them. So eventually, they will be able to take rational decisions because they understand the value of each state and do that, but that doesn't mean they

[9] How We Learn: Why Brains Learn Better Than Any Machine... for Now — book
[10] Surfaces and Essences: Analogy as the Fuel and Fire of Thinking — book

come up with this concept. So for a chess engine, it is that this move improves the value of this position. Right? But it doesn't necessarily transfer to other aspects of life. Right? So in in chess, for example, you could argue that controlling center means means controlling important or strategic positions. Right? You could you could once you learn this, you could also apply it to other aspects. Right? This is not just about 4 or 5 squares in the center of a chessboard. It's about importance of specific positions in different parts. So it could be about controlling an important road in a country, controlling an important passage. So that form of abstractions is what I think, and the way we store knowledge and learn is what allows us to scale very fast. Once we learn this in chess, we are able to scale it, like use it in across other domains. But systems like AlphaGo or AlphaZero are not able to do that because the war to them is just that, and they don't have this kind of abstract representations and knowledge. So I think I'm not saying that learning abstractions is necessarily the only way that you could build an intelligent system, but certainly, it's helping humans to scale way faster and improve faster than machines.

Tim Scarfe 00:42:50

On on the the matter of generalizing out out of distribution, so what we're starting to see is I mean, obviously, of thought is is used a lot, which means that you can take something which is trained on a distribution and you can kind of like manipulate. I suppose you're you're doing some kind of directed retrieval to create, you know, compositions in in some fixed way. So there's there's chain of thought prompting. Some people are doing program induction. And it certainly seems to be the case that with a few examples, maybe with chain of chain of thought, you can get a language model to induce a very rich and diverse set of programs. And many of those programs seemed to have some kind of abstraction. Another thing we're seeing is transduction, which is where you do some kind of active fine tuning. So you take the the test instances, and then you you kind of like modify the existing model a little bit so that you get generalization in this domain. So there's loads and loads of approaches that seem to be working quite well for making these things do better OOD. What do you think about that?[11]

[11] Chain-of-Thought Prompting Elicits Reasoning in Large Language Models — paper

Iman Mirzadeh 00:43:55

Yeah, Okay. So yes. So you could do like, there are lots of ways to explore this. Right? Like chain of thought, program synthesis, all these topics. So my my question, which I'm trying to think about, like, these days a lot about is that these all these systems, all these methods, all these directions are built on some assumptions. And the assumption is that there is something going on in the model, some kind of understanding that we need to improve that. Right? I'm not quite sure if right now there is that if that assumption is correct. Right? So I'm not saying that it's correct or incorrect. I'm not sure about it. So if the that assumption is incorrect, then all the things you are doing, chain of thoughts, all the other methods for improving this kind of reasoning may not be helpful. Right? So I'm like, what I'm thinking about like, is that like what like taking a step back and really asking this question, like, is there anything there, like beyond some interpolation between distributions? Right? If there is, then sure, we can continue and build up on top of that. If there's not, then what are we doing? Like, why are we trying to do all these cons like, methods that it's very difficult, by the way, like like because we are also doing it in a, like, kind of ad hoc way. There is no coherent model of what the transformer model is doing at scale. So it's very, very tricky, I think. So right now, I'm leaning towards that the models don't have a correct representation and the way we build these models by construction is limiting. So I don't really see how other like, extending this would be helpful. Like, there was an old example of, if you wanna land on moon, you can do it with an airplane. Right? So there there are 2 questions. If you have an airplane and you wanna land on moon, then you could improve like the speed of plane, you could make it lighter, you could like make it go faster or all these things, but it doesn't necessarily help you land on the moon. So you could also work on like how would like wings should look like? How would, like, the the the what kind of runway should we use to have a faster takeoff? All those things. But that doesn't necessarily help you with your goal. Right? So I'm saying, like I'm not saying that it is impossible, like, right now. I'm saying that, like, we should ask a question. Is this the right vessel? Is it like does it make sense? So, yeah, in general, it's I think we should take a step back and think harder on the problem.

Scaling Laws and the Connectionism-Symbolism Debate

Tim Scarfe 00:47:15

Onto scaling laws. So, you know, Noam Brown is talking about this this new test time scaling law. You've said that it's very very difficult to convince people that that scaling laws, you know, don't work because you'll always have a group of people that say, oh, you know, just over the next hill, just, you know, if we if we 10 x the parameters, 10 x the compute, then we'll finally get there. How do you convince people that they're wrong?

Iman Mirzadeh 00:47:40

Oh, yeah. I mean, I can't really convince because the issue with the scaling law is that if I spend millions or billions of dollar and train a model like, don't know, 10,000,000,000,000 parameters on 100,000,000,000,000 tokens and come and show you, hey, this is the model, and it can do even simple mathematics beyond what it's been trained on. Right? It can create new knowledge. And then someone can claim that, hey, this is not the right scale. Like, if you scale it to 50,000,000,000,000 parameters, there will something emerge. And then I because my argument is not theoretical, I can't say, no, you're wrong. And because I can't be I won't be able to train that model at 50,000,000,000,000 and spend a few billion dollars, I can't say you're wrong. So there will be always this debate that something will emerge. And by the way, I don't have issue with scaling, right, like scaling as in a sense that, like, I'm not saying that it will hurt or it won't be helpful at all. I'm saying, like, if you are thinking about the scaling, that you should focus on the slope of your scaling. You shouldn't care about where you are right now, right? So if you want to compare 2 different systems or 1 architecture with another, you shouldn't care about what happens at the 100,000,000,000,000 or will something emerge. You should say, look, how fast these models will be able to learn and do novel things. So right now, I think in that sense, the slope is near 0 for all the architectures and methods we have, but we can focus on the slope rather than the points and endpoints.[12]

Tim Scarfe 00:49:24

On the slope then, how how does the slope of human scaling compared to LLM scaling?

[12] Scaling Laws for Neural Language Models — paper

Iman Mirzadeh 00:49:29

Yeah. That's 1 of the interesting things. We won't we weren't able to do that, but it will be interesting to there I think there are many studies. I'm not an expert in the field, but there are many studies like showing that how humans versus models react to novel environments, right? So if you compare those, I think I don't have any number, but I think there are many studies showing that humans can adapt to new environment. They can learn new environment very easily, much easier than at least LLM. So I think the slope of human is faster, like better than the slope of LLM, but it could be that there would be a better species with better scaling, better slope, and I I look forward to it.

Tim Scarfe 00:50:22

Very cool. So, Iman, I think you would agree that we need to have better abstract world models. Right? We need to have better representations. How's that gonna work?

Iman Mirzadeh 00:50:31

So I think yeah. I mean, it to me, it looks nearly impossible to build an intelligence system that operates without without an abstract model of the environment and the world and knowledge. Right? But right now, there are many questions that needs to be answered before that and we don't even have like that 1 of the issues that I have with the literature right now and also including me is that we don't even have basic basic answers to these questions. So imagine you wanna learn a function. Right? So 1 way to represent that function is representing with table. Right? Like input x and output is y. Right? And then imagine there is another way of representing the function is representing it in a way like this is a polynomial, right, like y equals x squared plus some number, something like that, right? And we we don't know, like, right now, there is no objective measure, and we don't even have an answer to that which 1 is better, right? You might come and say, like, the the second 1 where it has a form, it's like it's abstract or like something like that. This 1 is better, but we don't know, like, why would someone claim this 1 is better than the other. Right? I also believe the second 1 is better, but we don't have a like, we couldn't quantify this. Right? The second 1 is certainly not like, someone can claim that the second 1 is compressed in a compressed form. You don't need a table. You just need a polynomial. But I don't think that's a right that's necessarily correct because in order to represent something with polynomial, it's not about the number of characters in a string. It's about concepts, right? So in order to define a polynomial, you need to understand what does a function mean. What does function x squared mean? X to the power of n. And then that requires understanding what does continuity mean?

What does a real number mean, like all these concepts. So if you wanna encode everything that it takes to reach to y equals x squared plus some number, then that may be even if being being larger than the table. Right? And then if your only goal, again, is achieving something, referring back to our previous discussion, then these 2 are equal, right? You give input and both of them give you output, and that 1 is even faster, you just have a lookup. So we don't have a way of comparing 2 representations. So if I wanna sum up is that in order to answer all the questions you ask, we need to start with understanding what like, how to compare 2 representations. Once we do have, then if you have a good measure of comparing 2 representations, then we can build on top of that. But we don't currently have that. Actually, I was talking to a friend like, how do we compare these 2 representation? And I said, like, I don't have a measure, but similar to the idea of like, at some point in time, someone someone came up with the idea of like formalizing the concept of generalization. Before like 200 years ago, like generalization wasn't a concept in a thing. It was a word, but so it says like 1 representation could be more beautiful than the other. Right? Like, a polynomial is more beautiful than a table, but we don't have any formal way of defining what constitutes something being more beautiful. But we are thinking on about these problems all the time. So

Tim Scarfe 00:54:34

I guess the the core here is whether deep in your bones you are a connectionist and you believe that it's possible in principle for this kind of thing to work. Because obviously, you've you've when we'll talk about your paper in a minute, you've designed a symbolic test to prove that certain types of reasoning can't happen in or don't happen in an in an LLM. In 1988, there was this famous connectionism critique by Fodor and Pylyshyn. And they argued that these systems don't have systematicity. They don't have composition or generalization. They don't have invertibilities. They can't, you know, explain why they did things. They can't cannibalize their knowledge in in the way that that you're talking about. But other folks like Smolensky, get his name right, in 1990, and certainly people like Benji, they argue that, yeah, it's not a problem. You know, all of this symbol use can just emerge, you know, at some level of complexity or whatever. So do you think that in the future we can make LLMs do this kind of symbolic reasoning?[13]

Iman Mirzadeh 00:55:30

I think when we are talking about symbols, like the the notion of symbol is not like, we have to define the notion of symbol. Like, is symbol something

[13] Tensor Product Variable Binding and the Representation of Symbolic Structures in Connectionist Systems — paper

concrete and predefined or something like, do you agree that maybe do you accept the, like, middle activation of a model a deep learning model can represent a symbol or not. So if you agree on that, that could be looked at as a formal symbol, then I think, yes, the model will be able to do that. Otherwise, if you think that symbol is like the the form of, like, separate external system, then no. I think I'm more of living towards like symbols being emerged as a part of computation inside it.

Tim Scarfe 00:56:25

Oh, that's very interesting. Yeah. So I suppose that there's a spectrum of use. I mean, that there are some people who think that just pure connectionism symbol use can emerge. There are people who are quite sort of like in the middle. I think I'm in that camp, is that we can build LLM systems. So maybe sort of agent LLMs that talk with each other or maybe neuro symbolic architectures where the LLMs can can use tools or some combination of of the above. But but you're you're in the camp that we could in principle just have a pure play neural network and it could do, you know, symbol use.

Iman Mirzadeh 00:57:00

Yeah. I mean, yeah. Right now, I don't have like any re I don't see it like in theory at least. Right? Like, I don't see in theory why such a system won't be able to create that kind of symbols theoretically. But, like, it could at some point in like, create symbols and do computations on those symbols. I'm not saying, like, this is the only way. We shouldn't have anything, like, not very experienced in that sense, but I don't have this kind of thing. The same goes with, like, other, like like when we when I discuss architectures, right? So I don't know if the, like, maybe transformer is the right architecture or not. But right now, in theory, at least given the context and assuming the model is generating token because it becomes true and complete, I don't have any reason in theory to believe that's a limitation right now. It could be that other architectures are better. Like, we can develop better architectures. It's just that right now, I don't see that as a limitation.

GSM-Symbolic: Exposing LLM Reasoning Fragility

Tim Scarfe 00:58:09

Okay. But but you do think we need to have Turing completeness for for symbol use. We could somehow come up with a neural network which which is Turing complete.

Iman Mirzadeh 00:58:17

Yeah.

Tim Scarfe 00:58:17

Okay. We're not there yet, but maybe.

Iman Mirzadeh 00:58:20

Yeah. Yeah. I think, like, yeah, we are when we're defining intelligence, like, at some point, like, Turing comp like, being able to do Turing complete is like operation is like a necessary condition, even.

Tim Scarfe 00:58:32

Yes. I completely agree with that. We should talk about your your GSM symbolic paper. So this is a is a landmark paper. It it did the rounds on on socials. Millions of people in my community said that I I had to interview you because it was amazing. Sketch it out for me.

Iman Mirzadeh 00:58:47

Oh, okay. So a little bit of background, like, we were working on understanding reasoning and we were exploring a few ideas about improving the reasoning of the models in terms of like improving the like increasing the amount of computation a model does per output token. And for that, we needed to evaluate the model and also be able to have a robust evaluation and then eventually have maybe better training data. So we started with the evaluation like creating a small sample from GSM symbolic just template, And then just to do a sanity check serial, we ran the experiments and see just like how close these numbers are to GSM symbol GSM 8 k, and we observed that there is a huge for some models, there is a very huge gap. For example, phi 2 had, like, 20% gap, 14% gap, something like that. And then there's also huge variation. Right? So, yeah, we we, like, kind of got sidetracked and, like, explored this immediately to understand, like, what's going on. Right? And 1 of 1 of the second like, after we had the initial GSM symbolic, the second thing I was working on was then GSM no op version, where you try to trick the model essentially and that you add 1 clause to the question where that clause carries no logical or like arithmetic operation. It's like if you completely ignore that clause, you will be fucked. Right? So that's why it's called GSM no op. And then that was the second 1. In the paper, we have a different presentation, but like we explore this second, and then we observed a very huge performance drop. So then we came back and try to understand like something in between. Like we try to be more like try to create like easier benchmarks than GSM no op by just adding 1 clause that carries operation and then adding 2 clauses that carries operation. So you can't ignore that and see how the model's performing that. And to like, among all those benchmarks and experiments that we designed, the the the surprising thing to me at least is that they did the variance of the performance we observe across the model. So I wish we could at least do some sort of cross evaluation across students of how students would perform if the if you change the me like the words in the problem without changing the numbers. Right? Like, if you instead of saying Amy has 3 apples, you can say John has 3 bananas. Right? They're the same, like, in terms of logical reasoning. And we I wish we could do that, And I think hope someone else explore these things, like how would human student perform if you just change orange to banana Yeah. Compared to an LLM. And so to me, the the the most fascinating part is the the variation in in the sense that why does this, like, gap exist even though you only change the numbers. So we have a benchmark called GSM names where we only change the proper names of the question. We don't change the numbers or add anything to the question. So yeah.

Tim Scarfe 01:02:43

The frontier model, I mean, to be honest, they still did quite well. I mean, I would expect them to just like drop off. Did you know, like, there's there's Subbarao Kambhampati's done some experiments with planning. And he he did this mystery blocks world where he just changed the name of the symbols to like random things and it dropped off a cliff. The o 1 model kind of like, you know, made it up to nontrivial performance again. But there was a huge decline. And I guess reading your paper, I was still somewhat intrigued that the frontier models, they did obviously like much worse, but but it was still pretty like nontrivial performance.

Iman Mirzadeh 01:03:20

There are 2 things. Like, personally, I I don't know. Like, I think overall, the GSM 8 k should be very simple benchmark given we we also there is also this factor that we don't exactly know on what kind of data these models have been trained. It could be. I'm not saying it is, but it could be the case that these models are also being trained on some sort of synthetic data generated from questions similar to math for problems like GSM 8 k. And we also know that the like, all the companies that are trying to build LLMs, they are also getting lots of human created question and answers in math and other domains. Right? So we don't exactly know. Like, that's 1 of the issues with not being, like, transparent about how what kind of data the at least the model's been trained on. But we it could be the case that the frontier models have, like, access to better data quality and data format of similar to GSM 8 k, and it could be that they have unlocked some kind of emergent ability. But if that were true, then at least I'm not a believer in benchmarks, but there are, again, different benchmarks that are similar to GSM HK, slightly more difficult. And in that those benchmarks, I think someone tried a similar idea that we did on GSM symbolic on math dataset, Hendrycks' MATH. And then they observed a larger performance drop for frontier models as well. But overall, it's again, for me, it's not about performance. It's not about like, I prefer a model that if you make the question difficult, the performance drops 10%, but there is no variation because if it understands the question, it understands the question to a model that drops, 1%, but there's a, like, large variation. So it's to me, it's not about the number of like like, the accuracy number. It's about why would a system that understands is able to do a set of logical steps tries to do differences of logical steps and gets it wrong if you change orange to banana in a question. Right? And that probably coming from because we train the system to learn a distribution. There is no concept of in this question, it doesn't matter what kind of object that is. It's it's about the number of objects. Right? So these 2 for us seems trivial, but for a system that is being trained to predict the exact object in the question during the training, it's nontrivial. So, yeah, I think overall in all these systems, I don't think they're truly understand this concept. Like, you

could change any like, I wasn't like doing a search on those kind of things, but you could like if you spend enough time any question that they get them wrong. I truly believe if you spend enough time, you could change those that question to a form that the models including frontier models gets and run. But unfortunately, the field looks at these things as a accuracy number, not necessarily what does it mean for a number to drop or what does it mean for a variation to increase. And they kind of now, I think the field is trying to move from GSM 8 k because the performance is reaching, like, 95%, near a 100%. And now I think, like, yesterday in the workshop, there was, like, oh, we have a new difficult benchmark called frontier math, and this has been designed so that's like now frontier models get 10%. Now let's increase that 10 to 90. I will I like, my problem with benchmarks is that they always is they are like a cut of reality. And once you freeze that, you could change the system in a way that's indirectly you change the system, but it has a direct, like, impact on the performance. So eventually, that benchmark will be always saturated, but it doesn't, at least to me, it doesn't matter. Like, that system, unless we change something fundamental, that system may get 99% on that benchmark, won't be able to create new knowledge, won't be able to understand what's going on. And to me, that's what matters. Right? So in image and computer vision, we had these benchmarks like ImageNet and all those benchmarks, and we saturated them. And we thought, the vision is solved, but now we see that self driving car is not becoming a thing right now because it's very difficult. Like, in real world, there isn't a specific frozen cut off reality that with fixed number of example. Reality will change. There will you you have to build an agent that understands and reason. So that's why I think it's important to focus on, not the exact number.

Tim Scarfe 01:08:38

Yes. And the the benchmarking is a big problem. So as as you say, I'm sure many of the frontier models have basically memorized GSM 8 k. And certainly, we should be moving towards more of a generative benchmark type thing where, you know, of course, it's not deterministic, but we have some generative system, and and we sample from it enough times, and we report some kind of average or something like that. But you you did show some some very interesting stuff. So first of all, that there was I think a change on the the no op data set where you actually 8 shot it, which means the model should be able to kind of filter away the distractors, but it didn't, which was very, very curious. But more broadly though, isn't it interesting that you you sample from these models a whole bunch of times and you see this huge variation? You know, and what's the implication? Is the implication that when we use language models for doing reasoning, we should actually be kind of like sampling a 100 times and taking the average result or something. Mean, because because many of us use language models and we just sample once. We just assume that, oh, it's doing reasoning. That's the right answer. I mean, we we don't really think about, well, if I asked it another 100 times, it would give me a bunch of different answers.[14]

Iman Mirzadeh 01:09:43

Yeah. I mean, so, yeah, I mean, about sampling, there are couple of things. Like, sometimes sampling in general doesn't make sense. So again, back to our example, 2 plus 2 equals, there's no sampling. It should be 4. Right? Like, if you increase the temperature, it could become 5, but why why do you wanna do that? Like, randomly picking a number doesn't make sense at all in that sense. That's why, yeah, for our study, we always stick to greedy decoding, no sampling. Because if you are doing arithmetics or you are doing reasoning, it has to be greedy. But in general, the other argument against sampling and majority voting and all those kind of thing is the example I remember, you had an episode where you there was an example, like, you let 10,000 drunken people go to after the bar, go to home, eventually, of them may reach their home, but that doesn't mean they are able to, like, understand what's going on. Like, given enough samples, you will eventually reach to destination probably, but that doesn't mean that's that's the issue. So, actually, there was a work that argued exactly the same thing. Like I said, like, if you measure the GSM 8 k and math accuracy and you sample the model 100 times, it will be, like, 20% better. But for the same argument, I don't think that's what we should study.

[14] GSM8K: Training Verifiers to Solve Math Word Problems — reference

Tim Scarfe 01:11:17

Cool. Well, Iman, it's been an absolute honor to have you on MLST. Thank you so much for joining us.

Iman Mirzadeh 01:11:22

Thank you. It was great talking to you.

References

[1] **Mastering Chess and Shogi by Self-Play with a General Reinforcement Learning Algorithm**
https://arxiv.org/abs/1712.01815
paper

[2] **Game Changer: AlphaZero's Groundbreaking Chess Strategies**
https://www.amazon.com/Game-Changer-AlphaZeros-Groundbreaking-Strategies/dp/9056918184
book

[3] **NLP Course: Language Modeling**
http://lena-voita.github.io/nlp_course/language_modeling.html
reference

[4] **GSM-Symbolic: Understanding the Limitations of Mathematical Reasoning in Large Language Models**
https://arxiv.org/abs/2410.05229
paper

[5] **Connectionism and Cognitive Architecture: A Critical Analysis**
https://www.sciencedirect.com/science/article/pii/001002779090014B
paper

[6] **On the Measure of Intelligence**
https://arxiv.org/abs/1911.01547
paper

[7] **On definition of intelligence**
https://www.sciencedirect.com/science/article/pii/S0160289624000266
paper

[8] **Defining Intelligence**
https://cis.temple.edu/~wangp/papers.html
paper

[9] **How We Learn: Why Brains Learn Better Than Any Machine... for Now**
https://www.amazon.com/How-We-Learn-Brains-Machine/dp/0525559884
book

[10] **Surfaces and Essences: Analogy as the Fuel and Fire of Thinking**
https://www.amazon.com/Surfaces-Essences-Analogy-Fuel-Thinking/dp/0465018475
book

[11] **Chain-of-Thought Prompting Elicits Reasoning in Large Language Models**

https://arxiv.org/abs/2201.11903
paper

[12] **Scaling Laws for Neural Language Models**
https://arxiv.org/abs/2001.08361
paper

[13] **Tensor Product Variable Binding and the Representation of Symbolic Structures in Connectionist Systems**
https://www.sciencedirect.com/science/article/abs/pii/000437029090007M
paper

[14] **GSM8K: Training Verifiers to Solve Math Word Problems**
https://huggingface.co/datasets/openai/gsm8k
reference