**CRAIG:** Hi, I’m Craig Smith and this is Eye on AI. Large Language Models are the rage these days but building them is mostly restricted to big organizations with plenty of resources. To most of us, they seem like Oz. But this week, Connor Leahy, one of the founders of the hacker collective, EleutherAI, pulls back to curtain to reveal that mere mortals – albeit very smart ones – can play in the same leagues as Google and OpenAI, given enough computational resources. Connor takes us into the guts of Large Language Models to show us around and speculate on the future toward which we are hurtling somewhat unawares.

**CONNOR:** I'm Connor Leahy. I am most well-known as one of the co-founders of the EleutherAI project. We describe ourselves as a decentralized research collective interested in open-source AI work and AI safety work. So, we're basically a bunch of weirdos hanging out in a chat room, doing research for fun.

**CONNOR:** We're most interested in AI work, especially large language models and similar large AI models that we think are really interesting science that currently has only really been done in large industry labs.

**CONNOR:** We wanted to get in on this. We want to do work with this as well. Literally the way this all started was me one day in an AI chat room saying, hey guys, why don't we give OpenAI, which is one of the largest players in this field, a run for their money.

**CONNOR:** Let's just mess around. Let's see how far we can get. Let's see, you know, can we build these things?

**CONNOR:** So nowadays we're a pretty big Discord server, like chat server with a relatively small, but still nontrivial core group. We're most well-known for ongoing efforts to build a very large GPT-3 size language model and open sourcing it.

**CONNOR:** It turns out very hard and very expensive to build these kinds of systems. Our currently largest release model is called GPT-J. To give a sense of scale here, GPT-3, which is considered one of the best models currently available from open AI, has about 175 billion parameters.

**CONNOR:** Just imagine more parameters equals better or smarter or whatever. Like it's not that easy obviously, but we can just, say it like that. Before GPT-3 came out, the largest models were in the range of like 1 billion parameters sometimes a few more.

**CONNOR:** And GPT-J is now at 6 billion parameters.

**CRAIG:** Can you give me a little bit of your background, where you went to school, what you currently do for a living and how large is the Discord group?

**CONNOR:** So, I'm basically mostly self-taught. I went to college for a few years in Munich, a lovely school, but dropped out because I was bored. So, I have no degree. Actually, most people that work at EleutherAI, have no degrees whatsoever. Were mostly self-taught hackers.

**CONNOR:** We consider ourselves a little bit of a descendant of the culture of the classic software hackers of the decades before us, just new field. The Discord server itself has I don't know, like 10,000 members, but that sounds more impressive than it is. Most people just lurk.

**CONNOR:** People that talk regularly, maybe like a hundred people or maybe 200, mostly discussing cutting-edge research. Every time a new paper comes out, it's instantly there and we immediately tear it apart and see if it's good or not. And the core group, like people really working on new applications and developing software maybe on the order of 10 people, 20 people.

**CRAIG:** One of the things I want to talk about is what's happening in China. Wudao 2.0 is now 10 times larger than GPT-3. Have you looked at Wudao 2.0, which I believe is open source? The code is on GitHub.

**CONNOR:** Wudao 2.0, is similar size to Google's largest model the Switch Transformer, which is like 1.6 trillion-ish parameters.

**CONNOR:** It is true that a lot of the Wudao source code or at least parts of it are online. I know that the FastMOE is, and I think CogView is like a few other bits and pieces. What is definitely not publicly available as the full model. So, the trained model, the trillion- parameter is not publicly accessible.

**CRAIG:** How did you go about building GPT-J. Of course, that's all built around a transformer algorithm. Transformers, there's a lot of open-source material.

**CRAIG:** And if you're involved in the field, you know how to write a transformer algorithm, you understand pre-training and there is plenty of data around. You need a certain amount of compute, which is not necessarily around for people without a lot of money. So, I'm just curious how easy you think it is, and easy as a relative term, obviously, given the amount that's in the research. Just tell me about that process.

**CONNOR:** GPT-J was trained using hardware that was provided to us as part of Google's TensorFlow Research Cloud Project, or TPU research cloud, which is like an academic access program where anyone can apply for access to do academic research or just open-source research. And they will give you access to some of their TPU's, which are similar to GPUs, but more specialized for neural network training, in order to train models and do experiments and such. So almost all of the credit, if not all of the credit, for GPT-J goes exclusively to Ben Wang who has done almost all of the work for GPT-J. Ben is amazing. He's at OpenAI now. And he basically wrote the whole code from scratch to run on TPUs because that's actually quite tricky, but he found a way to. He put in the effort to build a really good solid piece of code to train transformer models on TPUs of this size.

**CONNOR:** As for the difficulty, as models get larger, you need more compute to train them to completion. They get better, but you also need more computing power to train them to completion. And once models get big, once we're talking billions of parameters, it gets expensive, really fast.

**CONNOR:** So, while a model with maybe a hundred million parameters is something you might be able to train just with your local GPU that you bought for gaming, you might get away with that. Once these models get into the billion range, we're talking dozens or hundreds of things you want to run for weeks at a time to train these models.

**CONNOR:** And that just gets, more and more extreme as you go up. So, there's something of a discontinuity between, fits on your local machine to you need server grade hardware. And then there's this phase shift between one-ish billion or like a few billion, because then you can fit the model on one GPU and just run lots of copies.

**CONNOR:** And then things get really tricky when your model is so big that it doesn't fit into memory. So, you have to split the model. It's like several pieces and then put them on many different devices. This is one of the main things that GPT-J had to solve is implementing the scheme that you can split up the model onto multiple units and train it all at the same time.

**CONNOR:** This was very tricky at the time. There are more and more software libraries that make this easier and easier, but yeah, that's what you get at that area around the several billion level. And then for really big models, like a hundred billion parameters, something you run into a whole host of other issues.

**CONNOR:** So, the really interesting thing about training large models that we've learned at EleutherAI is that it's theoretically extremely straightforward. The math is understood. The algorithms are understood. it's all very well understood. But it is a lot of engineering effort. Things go wrong.

**CONNOR:** You have weird bugs, you have to optimize for performance very heavily, otherwise it's just never going to work. You run into very strange bugs with like numerical precision. They're like very hard to figure out what to do. And you play with a lot of numbers until you figure settings out and stuff.

**CONNOR:** So ultimately to train, an arbitrarily large model, everything is known on a theoretical level, but in practice it's very different. In practice this is pretty difficult. It's pretty tricky. As can be shown by a bunch of hackers in a cave figuring this out, it is definitely doable, but it gets harder the larger the models are.

**CRAIG:** Your model, for example, and then by extrapolation GPT-3 or Wudao, how many lines of code … I mean, when you talk about the parameters, essentially that's a measure of how many nodes in each layer.

**CONNOR:** A good way to think about a model is a box with a bunch of knobs that you can twiddle.

**CONNOR:** So, it's like numbers, you can turn up and down and the number parameters is how many such knobs you have that you can turn up and down. And the more knobs you have, the more things you can encode into the network. The more things you have, you can twiddle the more things you can change. So, you can imagine these models as being like, for example, GPT-3, to be a huge box of 175 billion knobs. And these knobs are then organized into layers and weights and biases. And each is one number that encodes one part of the computation that happens inside of the model.

**CRAIG:** And so, I've never actually asked these detailed questions because I'm not a coder.

**CRAIG:** A node consists of how many lines of code, and once you've written the code for one node is it a matter of just copying it or creating a loop in the program so that it runs through that node so many times.

**CONNOR:** So, the way you generally do it is, nowadays we have really good pre-written libraries by companies like Google and Facebook.

**CONNOR:** They have TensorFlow and PyTorch that make these things pretty easy. So, these are really complicated pieces of software that are like super optimized. So, with your code, what you define is a flow. You say, okay, first we have this many numbersand then we multiply them by the input numbers, and then we add this other number or something. And then when that's done, we do this with a whole new set of numbers and then repeat this like a hundred times. That's what's often called a layer so like often these models tend to be layered, you have one type of operation that's repeated over and over with different parameters each time.

**CONNOR:** It doesn't have to be that way, but it's just how these models tend to look. So, in terms of the actual code written by a human, you're usually nowadays on the order of like hundreds or thousands of lines. So really not that much. This is building on top of very complicated libraries.

**CONNOR:** I think amount of code that we at EleutherAI write, it's probably on the order more of thousands, maybe tens of thousands of lines for some of the more complex models we've built. The models themselves are often pretty simple. The complicated part is the training part that takes a lot of code sometimes to get that right.

**CONNOR:** But yeah, if you then want to make a larger model, you don't write additional code, generally. You just turn up a number and you say, okay, instead of 10 layers, we have a hundred, instead of the layer being a hundred units wide, it's 10,000 units wide or something. Usually, these are very easy to scale up and out.

**CONNOR:** Not always that easy, but usually.

**CRAIG:** And when you're adjusting the number of parameters or number of layers, it's just changing numbers in a line of code. It's not writing.

**CONNOR:** Yes.

**CRAIG:** Because you're talking about once a model gets to a certain size, it won't fit it memory. How do you measure that? Is that in terms of lines of code or what?

**CONNOR:** It's the parameters. So, these numbers you want to put on your GPU memory, cause you want to do operations on them. And your GPU has a certain amount of memory, 16 gigabytes or 40 gigabytes or whatever.

**CONNOR:** And once your model is bigger than that, you can't fit it on a single GPU anymore and you have to split into multiple GPU's. That wasn't really a thing a few years ago.

**CONNOR:** So, a lot of the older accelerators or not really optimized for this kind of training. Nvidia is really leading the pack here just incredibly.

**CONNOR:** They recently also bought Mellanox, which produces high inter-speed connectors. Because actually the interconnections, so, the speed between individual nodes in the computer, are the biggest bottleneck. The GPUs are often waiting for the network. So, you want as fast network as possible. And if you have the slightly slower than the best you can buy, it makes a huge difference. Actually, Nvidia was brilliant there that to train these very large models, having these really high-speed interconnects is really high-end hardware stuff is like super, super important.

**CRAIG:** So, let's talk about Wudao for a minute. They built this FastMOE algorithm to allow a scaling of the model. Did you use something similar or why is FastMOE important in going from 175 billion to 1.75 trillion, to that scaling.

**CRAIG:** Is it in the speed of the training? Is it in the amount of compute required?

**CONNOR:** So, comparing a MOE model, a mixture of expert model, to a non-MOE model is not a fair comparison. Let's call the non-MOE models, dense models for the moment. The way a normal dense model like GPT-3 works is that every time I give it a piece of data, I run through the whole model. I go through every parameter, I update every number, the whole model is used in every step.

**CONNOR:** What a MOE model does is that it instead splits the model into several experts. And then at each run through it selects a certain number of those experts to use and update. And it turns all the others off. So, each parameter in a MOE model is updated less often than a parameter is in a GPT model. What this means is that one dense parameter quote unquote counts for more in terms of performance than one MOE parameter, but MOE parameters are easier to train.

**CONNOR:** There's a lot of really nice things about MOE because, some of it can be turned off at different times, it's much easier to parallelize. You put some of the experts in some machines while the other ones are turned off. There's a lot of nice things you can do there. And I think it's pretty clearly a good direction for many types of these models. But it is misleading, I think, to the press.

**CONNOR:** The 1.6 trillion parameter switch transformer model, for example, Google showed performance was much lower than you would have expected from just scaling up a GPT-3 model to 1.6 trillion parameters.

**CONNOR:** Which was to be expected because MOE models need less compute to train, but they also converge to a less high performance. You use more memory; you have more parameters, but you need less compute because you don't update all the parameters all the time.

**CONNOR:** And of course, this has a nice side effect that it makes for a really great press release.

**CONNOR:** Having the big number looks really good, but all of these large models, you've seen, all of these trillion parameters, they're all MOE models. None of them are dense models.

**CRAIG:** What's the significance of these language models in your view? They're cool obviously, and Microsoft and open AI are trying to turn them into products. You have Copilot now, which is interesting, and maybe it'll become a commonly used tool, but what do you think is the significance of these large language?

**CONNOR:** I distinctly remember when I first saw GPT-3 blow up on Twitter and people showing off the things, I was genuinely, oh my God, this is the most important thing that’s ever happened in my life.

**CONNOR:** This is incredible. It's this thing that was trained just to reproduce text, just like a parrot. But it could talk to people. It could solve math equations. It could answer questions. It was never told to do any of these things. And I think people underestimate how massive of a deal that is. That was not obvious that this would work. One of the biggest problems in science is hindsight bias. We say something's impossible, then it happens and then we're like, oh no, we already knew that. But no, we did not know that this would happen. It is fully to open AI's credit to bravely have tried something crazy.

**CONNOR:** GPT was a crazy idea. I remember at the time everyone thought this was nuts and this is stupid. At the time, it was very much looked down upon to just scale up, quote, unquote dumb models.

**CONNOR:** Everyone's oh, we need smarter algorithms. We need more science, more causality, whatever the professor's pet interest is. But what's important about GPT-3, the real important, scientific discovery, about GPT-3 is not the model itself. The model's cool. And the text completion, that's fun.

**CONNOR:** Like all those things are cool. But what's really incredible about GPT-3 is that it is the same model as GPT-2, just larger, and learned all these things. That just by taking the same algorithm as a smaller model and just making the model larger, just giving it more data, allows it to unlock wholly new sets of skills without any human labeling or teaching or specific engineering.

**CONNOR:** You don't have to change the code. No one had to write a dialogue engine. It figured all this out by itself. That is incredible. I think that the scientific community is still not fully reckoned with how incredible it is that just by making numbers bigger just throwing more GPUs at it.

**CONNOR:** Whether or not it's intelligent depends on your philosophy of intelligence or whatever, but it is more of something. GPT-3 is definitely more of something than GPT-2 was or GPT-1. It knows, quote, unquote, something more. It has more skills. It can write better. And that was not because of some genius new algorithm or some clever new architecture.

**CONNOR:** It was just making it larger. And will there be a limit probably, but as far as I can tell, we're not there yet.

**CONNOR:** Will GPT-3 be commercially viable? No idea. It's incredibly expensive. You can do some fun things with it. Is it going to make a billion dollars? But some model that uses a scientific discovery we made with GPT-3, some model like that is going to be incredibly important in the near future.

**CRAIG:** Rich Sutton, the father of reinforcement learning, he's been saying for a while that the future is just scale that it's not new algorithms, and indeed that's, what's happening.

**CRAIG:** What impresses me about it as one it's unsupervised, it doesn't depend on labeled data, which is a huge constraint and two that it is multimodal or it crosses domains, at least between speech and vision and presumably, it will be able to handle other kinds of data. And that is a step towards more general AI.

**CONNOR:** So, you've already mentioned Richard Sutton. His famous essay called The Bitter Lesson which is that no matter how clever your algorithm is and how many PhD students you threw at it, at some point, it's just going to be out computed by a really dumb, simple algorithm with lots of compute.

**CONNOR:** I think the brain is probably just like a bunch of dumb algorithms thrown together by evolution to that just work and I don't expect our first like very powerful intelligences, whether you call them AGI or not, doesn't really matter. I expect at some point we're going to throw together a bunch of really big computers, lots of data, some simple, clever algorithms and it will learn, and it will be capable of doing many different things. Whether that requires multimodal data, or if transformers are enough or, if it works on GPUs those are all details. They don't really ultimately matter. Those are all solvable engineering problems.

**CONNOR:** Many of the biggest questions about how to build these systems have maybe not been solved, but we've clearly made big progress. And this progress continues into the future. I think it's pretty obvious that we're in for a weird time.

**CRAIG:** I'm interested in the growing competition. Between the U S and China, both at the research level but more importantly, between the governments and their militaries. Whether we like it or not governments and militaries harness new technology that comes from their research communities.

**CONNOR:** I think everyone in the AI field should be thinking about things like this a lot, because this is this important. It's going to become more important. AI is a general-purpose technology. It is power. It will allow you to solve problems and that is in itself ethically neutral. Having more power, being able to solve more things is neither good, nor bad. Depends on what problems you solve and what you define as a quote unquote problem. Different political groups may disagree on what they consider to be a problem that needs solving.

**CONNOR:** And only very recently have governments started to really wake up to how impactful these things might be in the near future.

**CONNOR:** Racing towards these kinds of technologies is very unfortunate and or dangerous.

**CONNOR:** Again, I don't think AI is evil. It's a technology like any other. Is nuclear power evil? No, it's a powerful technology, but it's not good or evil. It's just a thing. And I think AI is looking to be the nuclear power of the 21st century.

**CONNOR:** Powerful AI technologies will allow us to do incredible things, to speed up scientific progress to cure diseases, but the same technology, can also lead to very terrible outcomes and can do very damaging things.

**CONNOR:** But I think there's one very crucial way in which AI is different.

**CONNOR:** Remember these things are trained. We don't know how these things work internally. They're black boxes. There's just a bunch of numbers. We multiply the numbers and then something comes out of it. That's all we know. It's truly shocking how little we understand the algorithms we're playing with. We have to think of AIs as aliens. They're not human. They're like weird aliens that we've built or created of some kind. We don't know how they work, but they don't think like us. And they're very good at optimizing goals.

**CONNOR:** Genies are like AIs. The classic story of the genie is the genie grants you three wishes. And the third one is always to undo the first two. But we might not get that third wish because the first one might already blow us up.

**CONNOR:** So, my main concern about the Chinese American dynamic that is going on here is that we're both racing towards building a genie that's going to blow us up. And I think what we should be doing is all taking a step back and thinking very carefully, hey, hold on a second. Maybe we should understand these algorithms first, before we build them bigger and bigger or before we hook them up the weapons systems, or before we hook them up to the internet. Maybe we should study these systems and try to understand how we can control them and guarantee that they don't cause nuclear war or something like that, because that would be in their personal interest.

**CONNOR:** They are genies and how to control a genie is a really hard unsolved problem.

**CRAIG:** I don't think you're going to stop governments from pouring money into development, but they have to start talking to each other in some formal way to come to an agreement on a lot of these questions and not be working in parallel. But government's aside, here you are, a distributed unofficial group. If you can build a large language model, any group of like-minded scientists with whatever ideology they're pursuing, could build a large language model that could compete with governments.

**CRAIG:** The people behind Al Qaeda and ISIS, we're not idiots. There were a lot of very well-educated people involved in those groups. So how do you deal with that?

**CONNOR:** This is something I think about just about every day and I'm not super optimistic this is going to turn out okay. I think the chance that our species survives the century is not looking good.

**CONNOR:** It's not impossible, but it ain't looking good. Whether it's because of AI or something else. I think it's probably going to be AI, but it could also be something different. There's a law of mad science, which is every year, the minimum IQ needed to destroy the world drops by one.

**CONNOR:** This is not a new observation, by the way, even John von Newman and other people at the time, thought we should have a one world government and restrict what everyone can do because otherwise, people are going to build suitcase nukes and destroy the whole world.

**CONNOR:** It is funny how much 21st century is echoing the 20th century, just with higher stakes.

**CONNOR:** Obviously in a perfect world, we'd all just be friends.

**CONNOR:** We'd all shake hands and be like, okay, let's all work on this together as a species, let the whole planet come together and we'll all work on this together. No one does something in secret, no one tries to backstab the other people, everyone just gets together and we all - but that's never going to happen.

**CONNOR:** That was never going to happen. That's just not how humans work, unfortunately. Our species is very bad at these kinds of things. There are some things that we humans are really damn good at. Coordination at this scale is not one of them.

**CONNOR:** The way I see things going well is that we solve the technical genie problem, what's called the alignment problem, the control of the control problem. It's a no one's interest to build an uncontrollable AI, but I think people are going to be overconfident there could be overconfident and they're going to be in a race dynamic to be the first one there.

**CONNOR:** So, they're going to cut corners on research.

**CONNOR:** At EleutherAI our main motivation for building these models and releasing them is because we want more safety researchers, more alignment researchers to have access to this technology, to accelerate safety research. So, if we can solve the safety problems and it could be that this is one of the hardest problems humanity's ever faced, and we have to develop whole new kinds of philosophies of science to solve this. Maybe. We don't know. So far, it seems really hard. Professor Nick Bostrom at Oxford puts it very succinctly: alignment is like philosophy with a deadline. We have a deadline.

**CONNOR:** And if we don't figure this out, we're in a lot of shit. My hope is that by accelerating safety work, accelerating alignment work and theory in this direction, maybe we won't blow ourselves up accidentally.

**CRAIG:** Yeah. My thought is that AI models could be the ultimate optimization machine for global cooperation. Presumably an AI could come up with the perfect solution for the Chinese political system and culture to live in harmony with the U S or Western political system and culture and bring Russia into the fold.

**CRAIG:** But I'm a journalist, not a technologist.

**CONNOR:** Yeah. There's some kind of coordination we're very bad at, and that is coordination that AI might be very good at. AI can do types of coordination, like inspecting the source code of the other person that humans can't do.

**CONNOR:** So, AIs could prove to each other that they're not lying, for example. Imagine if we could do that, if I could prove that my political enemy is not hiding anything from me, he's not lying. Every single thing he says is the truth, what kind of coordination would that enable? We could makecontracts and deals and alliances that are so much more secure. That are so much safer than could ever be done between humans. But that still implies that we have AIs that we trust and that are on our side and that we can control.

**CRAIG:** There's a concern about unaffiliated groups or groups with troublesome ideologies building these kinds of models that could wreak havoc.

**CONNOR:** Yes. A bunch of hackers in a cave did get really far in this, but that's just because we got corporate sponsors. That's just because we had sponsors from very rich people that have a lot of access to hardware. If governments really want it to stop people from building large models, I think it is doable given the current levers, because it's just hard to hide a hundred-million-dollar supercomputer. If the government just slapped some kind of regulation on large machines like that, that would give them a lot of control over these kinds of things.

**CONNOR:** With smaller models, it's a bit different. A big open question is how good is the optimal algorithm? Some people I know think that it should be possible to run human level intelligence on a 1995 laptop. I don't think that's probably true, but some people think so. And they're smart people. Like they have reasons to believe this. If that's the case, that's going to be a fun timeline.

**CRAIG:** Not even a laptop. There's a distributed computing in the cloud. All you need is the cash, right?

**CONNOR:** Maybe. We don't know yet. Currently, it's actually, as someone who has tried to do this, it's actually very difficult to get the right hardware to train large language models.

**CONNOR:** It's not impossible, but like the number of companies that can do it is like dozens or hundreds, not thousands. And they all have very high capital investment. I can't open Amazon web services and just rent enough GPUs. No, I have to call up their people and I have to negotiate with them because you need a lot of them. They will probably sell it to me if I have enough cash, but it's a non-trivial step.

**CRAIG:** It's not something you could do anonymously.

**CONNOR:** Yeah. Maybe there are some shady dealers in some other countries, but currently that it's just not the case.

**CRAIG:** That’s it for this episode. If you want to read a transcript of our conversation today, you can find one on our website, eye-on.ai. If you want to play around with GPT-J, visit EleutherAI at [www.eleuther.ai](http://www.eleuther.ai) for links to the model and its code.