Lex Fridman Podcast #416 - Yann Lecun: Meta AI, Open Source, Limits of LLMs, AGI & the Future of AI

Published - March 7, 2024

Transcribed by - thepodtranscripts.com

Lex Fridman

The following is a conversation with Yann LeCun, his third time on this podcast. He is the chief AI scientist at Meta, professor at NYU, Turing Award winner and one of the seminal figures in the history of artificial intelligence. He and Meta AI have been big proponents of open sourcing, AI development and have been walking the walk by open sourcing many of their biggest models, including Llama 2 and eventually Llama 3. Also, Yann has been an outspoken critic of those people in the AI community who warn about the looming danger and existential threat of AGI. He believes the AGI will be created one day, but it will be good. It will not escape human control, nor will it dominate and kill all humans. At this moment of rapid AI development, this happens to be somewhat a controversial position, and so it's been fun seeing Yann get into a lot of intense and fascinating discussions online as we do in this very conversation. This is the Lex Fridman podcast. To support it, please check out our sponsors in the description. And now, dear friends, here's Yann LeCun. You've had some strong statements, technical statements about the future of artificial intelligence throughout your career actually, but recently as well, you've said that autoregressive LLMs are not the way we're going to make progress towards superhuman intelligence. These are the large language models like GPT-4, like Llama 2 and 3 soon and so on. How do they work and why are they not going to take us all the way?

Yann LeCun

For a number of reasons. The first is that there is a number of characteristics of intelligent behavior. For example, the capacity to understand the world, understand the physical world, the ability to remember and retrieve things, persistent memory, the ability to reason, and the ability to plan. Those are four essential characteristics of intelligent systems or entities, humans, animals. LLMs can do none of those or they can only do them in a very primitive way and they don't really understand the physical world. They don't really have persistent memory. They can't really reason and they certainly can't plan. And so if you expect the system to become intelligent just without having the possibility of doing those things, you're making a mistake. That is not to say that autoregressive LLMs are not useful. They're certainly useful, that they're not interesting, that we can't build a whole ecosystem of applications around them. Of course we can, but as a pass towards human-level intelligence, they're missing essential components. And then there is another tidbit or fact that I think is very interesting. Those LLMs are trained on enormous amounts of texts, basically, the entirety of all publicly available texts on the internet, right? That's typically on the order of 10 to the 13 tokens. Each token is typically two bytes, so that's two 10 to the 13 bytes as training data. It would take you or me 170,000 years to just read through this at eight hours a day. So it seems like an enormous amount of knowledge that those systems can accumulate, but then you realize it's really not that much data. If you talk to developmental psychologists and they tell you a four-year-old has been awake for 16,000 hours in his or her life, and the amount of information that has reached the visual cortex of that child in four years is about 10 to 15 bytes. And you can compute this by estimating that the optical nerve carry about 20 megabytes per second roughly, and so 10 to the 15 bytes for a four-year-old versus two times 10 to the 13 bytes for 170,000 years worth of reading. What that tells you is that through sensory input, we see a lot more information than we do through language, and that despite our intuition, most of what we learn and most of our knowledge is through our observation and interaction with the real world, not through language. Everything that we learn in the first few years of life, and certainly everything that animals learn has nothing to do with language.

Lex Fridman
So it would be good to maybe push against some of the intuition behind what you're saying. So it is true there's several orders of magnitude more data coming into the human mind much faster, and the human mind is able to learn very quickly from that, filter the data very quickly. Somebody might argue your comparison between sensory data versus language, that language is already very compressed. It already contains a lot more information than the bytes it takes to store them if you compare it to visual data. So there's a lot of wisdom and language. There's words, and the way we stitch them together, it already contains a lot of information. So is it possible that language alone already has enough wisdom and knowledge in there to be able to, from that language, construct a world model and understanding of the world, an understanding of the physical world that you're saying LLMs lack?

Yann LeCun
So it's a big debate among philosophers and also cognitive scientists, like whether intelligence needs to be grounded in reality. I'm clearly in the camp that yes, intelligence cannot appear without some grounding in some reality. It doesn't need to be physical reality. It could be simulated, but the environment is just much richer than what you can express in language. Language is a very approximate representation or percepts and/or mental models. I mean, there's a lot of tasks that we accomplish where we manipulate a mental model of the situation at hand, and that has nothing to do with language. Everything that's physical, mechanical, whatever, when we build something, when we accomplish a task, model task of grabbing something, etc. – we plan or action sequences, and we do this by essentially imagining the result of the outcome of a sequence of actions that we might imagine and that requires mental models that don't have much to do with language, and I would argue most of our knowledge is derived from that interaction with the physical world. So a lot of my colleagues who are more interested in things like computer vision are really on that camp that AI needs to be embodied essentially. And then other people coming from the NLP side or maybe some other motivation don't necessarily agree with that, and philosophers are split as well, and the complexity of the world is hard to imagine. It's hard to represent all the complexities that we take completely for granted in the real world that we don't even imagine require intelligence, right? This is the old Moravec paradox, from the pioneer of robotics, hence Moravec, who said, how is it that with computers, it seems to be easy to do high-level complex tasks like playing chess and solving integrals and doing things like that, whereas the thing we take for granted that we do every day, like, I don't know,

learning to drive a car or grabbing an object, we can't do with computers, and we have LLMs that can pass the bar exam, so they must be smart, but then they can't learn to drive in 20 hours like any 17-year old, they can't learn to clear out the dinner table and fill up the dishwasher like any 10-year old can learn in one shot. Why is that? What are we missing? What type of learning or reasoning architecture or whatever are we missing that basically prevent us from having level five sort of in cars and domestic robots?

Lex Fridman
Can a large language model construct a world model that does know how to drive and does know how to fill a dishwasher, but just doesn't know how to deal with visual data at this time, so it can operate in a space of concepts?

Yann LeCun
So yeah, that's what a lot of people are working on. So the short answer is no, and the more complex answer is you can use all kinds of tricks to get an LLM to basically digest visual representations of images or video or audio for that matter. And a classical way of doing this is you train a vision system in some way, and we have a number of ways to train vision systems either supervised, semi-supervised, self-supervised, all kinds of different ways, that will turn any image into a high-level representation. Basically a list of tokens that are really similar to the kind of tokens that typical LLM takes as an input. And then you just feed that to the LLM in addition to the text, and you just expect the LLM, during training, to be able to use those representations to help make decisions. I mean, there's been work along those lines for quite a long time and now, you see those systems. I mean there are LLMs that have some vision extension, but they're basically hacks in the sense that those things are not trained to really understand the world. They're not trained with video, for example. They don't really understand intuitive physics, at least not at the moment.

Lex Fridman
So you don't think there's something special to you about intuitive physics, about sort of common sense reasoning about the physical space, about physical reality. That to you is a giant leap that LLMs are just not able to do?

Yann LeCun
We're not going to be able to do this with the type of LLMs that we are working with today, and there's a number of reasons for this, but the main reason is the way LLMs are trained is that you take a piece of text, you remove some of the words in that text, you mask them, you replace them by blank markers, and you train a genetic neural net to predict the words that are missing. And if you build this neural net in a particular way so that it can only look at words that are to the left or the one it's trying to predict, then what you have is a system that basically is trying to predict the next word in a text. So then you can feed it a text, a prompt, and you can ask it to predict the next word. It can never predict the next word exactly. So what it's going to do is produce a probability distribution of all the possible

words in a dictionary. In fact, it doesn't predict words. It predicts tokens that are kind of subword units, and so it's easy to handle the uncertainty in the prediction there because there is only a finite number of possible words in the dictionary, and you can just compute a distribution over them. Then what the system does is that it picks a word from that distribution. Of course, there's a higher chance of picking words that have a higher probability within that distribution. So you sample from that distribution to actually produce a word, and then you shift that word into the input, and so that allows the system not to predict the second word, and once you do this, you shift it into the input, etc. That's called autoregressive prediction, which is why those LLMs should be called autoregressive LLMs, but we just call them LLMs, and there is a difference between this kind of process and a process by which before producing a word - When you and I talk, you and I are bilingual, we think about what we're going to say, and it's relatively independent of the language in which we're going to say. When we talk about, I don't know, let's say a mathematical concept or something, the kind of thinking that we're doing and the answer that we're planning to produce is not linked to whether we're going to see it in French or Russian or English.

Lex Fridman
Chomsky just rolled his eyes, but I understand, so you're saying that there's a bigger abstraction that goes before language and maps onto language?

Yann LeCun
Right. It's certainly true for a lot of thinking that we do.

Lex Fridman
Is that obvious that we don't - you're saying your thinking is same in French as it is in English?

Yann LeCun
Yeah, pretty much.

Lex Fridman
Pretty much or how flexible are you if there's a probability distribution?

Yann LeCun
Well, it depends what kind of thinking, right? If it's producing puns, I get much better in French than English about that, or much worse.

Lex Fridman
Is there an abstract representation of puns? Is your humor an abstract - when you tweet and your tweets are sometimes a little bit spicy, is there an abstract representation in your brain of a tweet before it maps onto English?

Yann LeCun
There is an abstract representation of imagining the reaction of a reader to that text.

Lex Fridman
Or you start with laughter and then figure out how to make that happen?

Yann LeCun
Or figure out like a reaction you want to cause and then figure out how to say it so that it causes that reaction. But that's really close to language. But think about a mathematical concept or imagining something you want to build out of wood or something like this. The kind of thinking you're doing has absolutely nothing to do with language really. It's not like you have necessarily an internal monologue in any particular language. You are imagining mental models of the thing. I mean, if I ask you to imagine what this water bottle will look like if I rotate it 90 degrees, that has nothing to do with language. And so clearly, there is a more abstract level of representation in which we do most of our thinking, and we plan what we're going to say if the output is uttered words as opposed to an output being muscle actions, we plan our answer before we produce it. LLMs don't do that. They just produce one word after the other instinctively if you want. It's a bit like the subconscious actions where you're distracted, you're doing something, you're completely concentrated, and someone comes to you and asks you a question and you kind of answer the question. You don't have time to think about the answer, but the answer is easy. So you don't need to pay attention. You sort of respond automatically. That's kind of what an LLM does. It doesn't think about its answer really. It retrieves it because it's accumulated a lot of knowledge. So it can retrieve some things, but it's going to just spit out one token after the other without planning the answer.

Lex Fridman
But you're making it sound just one token after the other. One token at a time generation is bound to be simplistic, but if the world model is sufficiently sophisticated that one token at a time, the most likely thing it generates is a sequence of tokens is going to be a deeply profound thing.

Yann LeCun
But then that assumes that those systems actually possess an eternal world model.

Lex Fridman
So really goes to the- I think the fundamental question is can you build a really complete world model, not complete, but one that has a deep understanding of the world?

Yann LeCun
Yeah. So can you build this first of all by prediction, and the answer is probably yes. Can you build it by predicting words? And the answer is most probably no, because language is very

poor in terms of weak or low bandwidth if you want, there's just not enough information there. So building world models means observing the world and understanding why the world is evolving the way it is, and then the extra component of a world model is something that can predict how the world is going to evolve as a consequence of an action you might take. So one model really is here is my idea of the state of the world at time, T, here is an action I might take. What is the predicted state of the world at time, T+1? Now that state of the world does not need to represent everything about the world, it just needs to represent enough that's relevant for this planning of the action, but not necessarily all the details. Now, here is the problem. You're not going to be able to do this with generative models. So a generative model has trained on video, and we've tried to do this for 10 years, you take a video, show a system, a piece of video, and then ask you to predict the reminder of the video, basically predict what's going to happen.

Lex Fridman
One frame at a time, do the same thing as the autoregressive LLMs do, but for video.

Yann LeCun
Right. Either one frame at a time-

Lex Fridman
LVMs.

Yann LeCun
- or a group of frames at a time. But yeah, a large video model if you want. The idea of doing this has been floating around for a long time and at FAIR, some of our colleagues and I have been trying to do this for about 10 years, and you can't really do the same trick as with LLMs because LLMs, as I said, you can't predict exactly which word is going to follow a sequence of words, but you can predict the distribution of words. Now, if you go to video, what you would have to do is predict the distribution of all possible frames in a video, and we don't really know how to do that properly. We do not know how to represent distributions over high-dimensional, continuous spaces in ways that are useful. And there lies the main issue, and the reason we can do this is because the world is incredibly more complicated and richer in terms of information than text. Text is discrete, video is high-dimensional and continuous. A lot of details in this. So if I take a video of this room and the video is a camera panning around, there is no way I can predict everything that's going to be in the room as I pan around. The system cannot predict what's going to be in the room as the camera is panning. Maybe it's going to predict this is a room where there's a light and there is a wall and things like that. It can't predict what the painting of the wall looks like or what the texture of the couch looks like. Certainly not the texture of the carpet. So there's no way I can predict all those details. So one way to possibly handle this, which we've been working for a long time, is to have a model that has what's called a latent variable. And the latent variable is fed to a neural net, and it's supposed to represent all the information about the

world that you don’t perceive yet, and that you need to augment the system for the prediction to do a good job at predicting pixels, including the fine texture of the carpet and the couch and the painting on the wall. That has been a complete failure essentially. And we’ve tried lots of things. We tried just straight neural nets, we tried GANs, we tried VAEs, all kinds of regularized auto encoders. We tried many things. We also tried those kinds of methods to learn good representations of images or video that could then be used as input to, for example, an image classification system. That also has basically failed. All the systems that attempt to predict missing parts of an image or video from a corrupted version of it, basically, so take an image or a video, corrupt it or transform it in some way, and then try to reconstruct the complete video or image from the corrupted version, and then hope that internally, the system will develop good representations of images that you can use for object recognition, segmentation, whatever it is. That has been essentially a complete failure and it works really well for text. That’s the principle that is used for LLMs, right?

Lex Fridman
So where’s the failure exactly? Is it that it’s very difficult to form a good representation of an image, like a good embedding of all the important information in the image? Is it in terms of the consistency of image to image, to image to image that forms the video? If we do a highlight reel of all the ways you failed, what’s that look like?

Yann LeCun
Okay, so the reason this doesn’t work is first of all, I have to tell you exactly what doesn’t work because there is something else that does work. So the thing that does not work is training the system to learn representations of images by training it to reconstruct a good image from a corrupted version of it, okay? That’s what doesn’t work. And we have a whole slew of techniques for this that are variant of denoising autoencoders, something called MAE developed by some of my colleagues at FAIR, masked autoencoder. So it’s basically like the LLMs or things like this where you train the system by corrupting texts except you corrupt images, you remove patches from it, and you train a gigantic neural network reconstruct. The features you get are not good, and you know they’re not good because if you now train the same architecture, but you train it to supervise with label data, with textual descriptions of images, etc. - you do get good representations and the performance on recognition tasks is much better than if you do this self-supervised retraining.

Lex Fridman
The architecture is good?

Yann LeCun
The architecture is good, the architecture of the encoder is good, but the fact that you train the system to reconstruct images does not lead it to produce to long, good generic features of images.

Lex Fridman
When you train in a self-supervised way?

Yann LeCun
Self-supervised by reconstruction.

Lex Fridman
Yeah, by reconstruction.

Yann LeCun
Okay, so what's the alternative? The alternative is joint embedding.

Lex Fridman
What is joint embedding? What are these architectures that you're so excited about?

Yann LeCun
Okay, so now instead of training a system to encode the image and then training it to reconstruct the full image from a corrupted version, you take the full image, you take the corrupted or transformed version, you run them both through encoders, which in general, are identical, but not necessarily. And then you train a predictor on top of those encoders to predict the representation of the full input from the representation of the corrupted one. So joint embedding, because you're taking the full input and the corrupted version or transformed version, run them both through encoders, you get a joint embedding, and then you're saying, can I predict the representation of the full one from the representation of the corrupted one? And I call this a JEPA, so that means joint embedding predictive architecture because this joint embedding and there is this predictor that predicts the representation of the good guy from the bad guy. And the big question is how do you train something like this? And until five years ago or six years ago, we didn't have particularly good answers for how you train those things except for one, called contrastive learning, where the idea of contrastive learning is you take a pair of images that are, again, an image and a corrupted version or degraded version somehow or transformed version of the original one, and you train the predicted representation to be the same as that. If you only do this, this system collapses. It basically completely ignores the input and produces representations that are constant. So the contrastive methods avoid this, and those things have been around since the early '90s, I had a paper on this in 1993, is you also show pairs of images that you know are different, and then you push away the representations from each other. So you say, not only do representations of things that we know are the same should be the same or should be similar, but representation of things that we know are different should be different. And that prevents the collapse, but it has some limitation. And there's a whole bunch of techniques that have appeared over the last six, seven years that can revive this type of method, some of them from FAIR, some of them from Google and other places, but there are limitations to those contrastive methods. What has changed in the last three,

four years is now we have methods that are non-contrastive. So they don't require those negative contrastive samples of images that we know are different. You turn them on you with images that are different versions or different views of the same thing, and you rely on some other tricks to prevent the system from collapsing. And we have half a dozen different methods for this now.

Lex Fridman
So what is the fundamental difference between joint embedding architectures and LLMs? Can JEPA take us to AGI? Whether we should say that you don't like the term AGI, and we'll probably argue I think every single time I've talked to you, we've argued about the G in AGI.

Yann LeCun
Yes.

Lex Fridman
I get it. I get it. Well, we'll probably continue to argue about it. It's great. You like AMI because you like French and ami is friend in French, and AMI stands for advanced machine intelligence. But either way, can JEPA take us to that towards that advanced machine intelligence?

Yann LeCun
Well, so it's a first step. Okay, so first of all, what's the difference with generative architectures like LLMs? So LLMs or vision systems that are trained by reconstruction generate the inputs. They generate the original input that is non-corrupted, non-transformed, so you have to predict all the pixels, and there is a huge amount of resources spent in the system to actually predict all those pixels, all the details. In a JEPA, you're not trying to predict all the pixels, you're only trying to predict an abstract representation of the inputs. And that's much easier in many ways. So what the JEPA system, when it's being trained, is trying to do is extract as much information as possible from the input, but yet only extract information that is relatively easily predictable. So there's a lot of things in the world that we cannot predict. For example, if you have a self-driving car driving down the street or road, there may be trees around the road and it could be a windy day. So the leaves on the tree are kind moving in kind semi-chaotic, random ways that you can't predict and you don't care, you don't want to predict. So what you want is your encoder to basically eliminate all those details. It'll tell you there's moving leaves, but it's not going to give the details of exactly what's going on. And so when you do the prediction in representation space, you're not going to have to predict every single pixel of every leaf. And that not only is a lot simpler, but also, it allows the system to essentially learn an abstract representation of the world where what can be modeled and predicted is preserved and the rest is viewed as noise and eliminated by the encoder. So it lifts the level of abstraction of the representation. If you think about this, this is something we do absolutely all the time. Whenever we describe a phenomenon, we describe it at a particular

level of abstraction. We don't always describe every natural phenomenon in terms of quantum field theory. That would be impossible. So we have multiple levels of abstraction to describe what happens in the world, starting from quantum field theory, to atomic theory and molecules and chemistry, materials and all the way up to concrete objects in the real world and things like that. So we can't just only model everything at the lowest level. And that's what the idea of JEPA is really about, learn abstract representation in a self-supervised manner, and you can do it hierarchically as well. So that, I think, is an essential component of an intelligent system. And in language, we can get away without doing this because language is already to some level abstract and already has eliminated a lot of information that is not predictable. And so we can get away without doing the joint embedding, without lifting the abstraction level and by directly predicting words.

Lex Fridman
So joint embedding, it's still generative, but it's generative in this abstract representation space?

Yann LeCun
Yeah.

Lex Fridman
And you're saying language, we were lazy with language because we already got the abstract representation for free, and now we have to zoom out, actually think about generally intelligent systems. We have to deal with a full mess of physical reality, of reality. And you do have to do this step of jumping from the full, rich, detailed reality to a abstract representation of that reality based on what you can then reason and all that kind of stuff.

Yann LeCun
Right. And the thing is those self-supervised algorithm that learn by prediction, even in representation space, they learn more concept if the input data you feed them is more redundant. The more redundancy there is in the data, the more they're able to capture some internal structure of it. And so there is way more redundancy in the structure in perceptual inputs, sensory input like vision than there is in text, which is not nearly as redundant. This is back to the question you were asking a few minutes ago. Language might represent more information really, because it's already compressed. You're right about that, but that means it's also less redundant, and so self-supervision, you will not work as well.

Lex Fridman
Is it possible to join the self-supervised training on visual data and self-supervised training on language data? There is a huge amount of knowledge, even though you talk down about those 10 to the 13 tokens. Those 10 to the 13 tokens represent the entirety-Those 10 to the 13 tokens represent the entirety, a large fraction of what us humans have figured out, both the

shit-talk on Reddit and the contents of all the books and the articles and the full spectrum of human intellectual creation. So is it possible to join those two together?

Yann LeCun
Well, eventually, yes. But I think if we do this too early, we run the risk of being tempted to cheat. And in fact, that's what people are doing at the moment with vision-language model. We're basically cheating. We're using language as a crutch to help the deficiencies of our vision systems to learn good representations from images and video. And the problem with this is that we might improve our language models by feeding them images, but we're not going to get to the level of even the intelligence or level of understanding of the world of a cat or a dog, which doesn't have language. They don't have language and they understand the world much better than any LLM. They can plan really complex actions and imagine the result of a bunch of actions. How do we get machines to learn that before we combine that with language? Obviously if we combine this with language, this is going to be a winner, but before that, we have to focus on how do we get systems to learn how the world works?

Lex Fridman
So this joint-embedding predictive architecture, for you, that's going to be able to learn something like common sense, something like what a cat uses to predict how to mess with its owner most optimally by knocking over a thing.

Yann LeCun
That's the hope. In fact, the techniques we're using are non-contrastive. So not only is the architecture non-generative, the learning procedures we are using are non-contrastive. We have two sets of techniques. One set is based on distillation, and there's a number of methods that use this principle, one by DeepMind called BYOL, a couple by FAIR, one called vcREG and another one called I-JEPA. And vcREG, I should say, is not a distillation method actually, but I-JEPA and BYOL certainly are. And there's another one also called DINO or DINO also produced from at FAIR. And the idea of those things is that you take the full input, let's say an image, you run it through an encoder, produces a representation, and then you corrupt that input or transform it, run it through essentially what amounts to the same encoder with some minor differences and then train a predictor. Sometimes a predictor is very simple, sometimes it doesn't exist, but train a predictor to predict a representation of the first uncorrupted input from the corrupted input. But you only train the second branch. You only train the part of the network that is fed with the corrupted input. The other network, you don't train. But since they share the same weight, when you modify the first one, it also modifies the second one. And with various tricks, you can prevent the system from collapsing with the collapse of the type I was explaining before, where the system basically ignores the input. So that works very well. The two techniques we developed at FAIR, DINO and I-JEPA work really well for that.

Lex Fridman
So what kind of data are we talking about here?

Yann LeCun
So there's several scenario, one scenario is you take an image, you corrupt it by changing the cropping, for example, changing the size a little bit, maybe changing the orientation, blurring it, changing the colors, doing all kinds of horrible things to it.

Lex Fridman
But basic horrible things?

Yann LeCun
Basic horrible things that sort of degrade the quality a little bit and change the framing, crop the image. And in some cases, in the case of I-JEPA, you don't need to do any of this, you just mask some parts of it. You just basically remove some regions, like a big block essentially, and then run through the encoders and train the entire system, encoder and predictor, to predict the representation of the good one from the representation of the corrupted one. So that's the I-JEPA. It doesn't need to know that it's an image for example, because the only thing it needs to know is how to do this masking. Whereas with DINO, you need to know it's an image because you need to do things like geometry transformation and blurring and things like that, that are really image specific. A more recent version of this that we have is called V-JEPA. So it's basically the same idea as I-JEPA except it's applied to video. So now you take a whole video and you mask a whole chunk of it. And what we mask is actually kind of a temporal tube, so a whole segment of each frame in the video over the entire video.

Lex Fridman
And that tube was statically positioned throughout the frames, just literally it's a straight tube.

Yann LeCun
The tube, yeah, typically is 16 frames or something, and we mask the same region over the entire 16 frames. It's a different one for every video obviously. And then again, train that system so as to predict the representation of the full video from the partially masked video. And that works really well. It's the first system that we have that learns good representations of video so that when you feed those representations to a supervised classifier head, it can tell you what action is taking place in the video with pretty good accuracy. So that's the first time we get something of that quality.

Lex Fridman
That's a good test that a good representation is formed. That means there's something to this.

Yann LeCun
Yeah. We also preliminary result that seem to indicate that the representation allow our system to tell whether the video is physically possible or completely impossible, because some object disappeared or an object suddenly jumped from one location to another or changed shape or something.

Lex Fridman
So it's able to capture some physics based constraints about the reality represented in the video, about the appearance and the disappearance of objects.

Yann LeCun
Yeah, that's really new.

Lex Fridman
Okay, but can this actually get us to this kind of world model that understands enough about the world to be able to drive a car?

Yann LeCun
Possibly, this is going to take a while before we get to that point. And there are systems already robotic systems, that are based on this idea. And what you need for this is a slightly modified version of this, where imagine that you have a complete video and what you're doing to this video is that you are either translating it in time towards the future. So you only see the beginning of the video, but you don't see the latter part of it that is in the original one, or you just mask the second half of the video, for example. And then you train a JEPA system or the type I described, to predict the representation of the full video from the shifted one. But you also feed the predictor with an action. For example, the wheel is turned 10 degrees to the right or something, right? So if it's a dash cam in a car and you know the angle of the wheel, you should be able to predict to some extent what's going to happen to what you see. You're not going to be able to predict all the details of objects that appear in the view obviously, but at a abstract representation level, you can probably predict what's going to happen. So now what you have is a internal model that says, "Here is my idea of the state of the world at time T. Here is an action I'm taking. Here is a prediction of the state of the world at time T plus one, T plus delta T, T plus two seconds," whatever it is. If you have a model of this type, you can use it for planning. So now you can do what LLMs cannot do, which is planning what you're going to do. So as you arrive at a particular outcome or satisfy a particular objective. So you can have a number of objectives. I can predict that if I have an object like this and I open my hand, it's going to fall. And if I push it with a particular force on the table, it's going to move. If I push the table itself, it's probably not going to move with the same force. So we have this internal model of the world in our mind, which allows us to plan sequences of actions to arrive at a particular goal. And so now if you have this world model, we can imagine a sequence of actions, predict what the outcome of the sequence of action is going to be, measure to what extent the final state satisfies a particular objective, like

moving the bottle to the left of the table and then plan a sequence of actions that will minimize this objective, at runtime. We're not talking about learning, we're talking about inference time, so this is planning, really. And in optimal control, this is a very classical thing. It's called model predictive control. You have a model of the system you want to control that can predict the sequence of states corresponding to a sequence of commands. And you're planning a sequence of commands so that according to your role model, the end state of the system will satisfy an objectives that you fix. This is the way rocket trajectories have been planned since computers have been around, so since the early '60s essentially.

Lex Fridman
So yes, for a model predictive control, but you also often talk about hierarchical planning. Can hierarchical planning emerge from this somehow?

Yann LeCun
Well, so no, you will have to build a specific architecture to allow for hierarchical planning. So hierarchical planning is absolutely necessary if you want to plan complex actions. If I want to go from, let's say from New York to Paris, it's the example I use all the time, and I'm sitting in my office at NYU, my objective that I need to minimize is my distance to Paris. At a high level, a very abstract representation of my location, I would have to decompose this into two sub goals. First one is go to the airport, second one is catch a plane to Paris. Okay, so my sub goal is now going to the airport. My objective function is my distance to the airport. How do I go to the airport where I have to go in the street and hail a taxi, which you can do in New York. Okay, now I have another sub goal go down on the street. Well that means going to the elevator, going down the elevator, walk out the street. How do I go to the elevator? I have to stand up from my chair, open the door in my office, go to the elevator, push the button. How do I get up for my chair? You can imagine going down, all the way down, to basically what amounts to millisecond by millisecond muscle control. And obviously you're not going plan your entire trip from New York to Paris in terms of millisecond by millisecond muscle control. First, that would be incredibly expensive, but it will also be completely impossible because you don't know all the conditions of what's going to happen, how long it's going to take to catch a taxi or to go to the airport with traffic. I mean, you would have to know exactly the condition of everything to be able to do this planning and you don't have the information. So you have to do this hierarchical planning so that you can start acting and then sort of replanning as you go. And nobody really knows how to do this in AI. Nobody knows how to train a system to learn the appropriate multiple levels of representation so that hierarchical planning works.

Lex Fridman
Does something like that already emerge? So can you use an LLM, state-of-the-art LLM, to get you from New York to Paris by doing exactly the kind of detailed set of questions that you just did, which is, can you give me a list of 10 steps I need to do, to get from New York to Paris? And then for each of those steps, can you give me a list of 10 steps, how I make that

step happen? And for each of those steps, can you give me a list of 10 steps to make each one of those, until you're moving your individual muscles, maybe not, whatever you can actually act upon using your own mind.

Yann LeCun
Right. So there's a lot of questions that are also implied by this, right? So the first thing is LLMs will be able to answer some of those questions down to some level of abstraction, under the condition that they've been trained with similar scenarios in their training set.

Lex Fridman
They would be able to answer all of those questions, but some of them may be hallucinated meaning non-factual.

Yann LeCun
Yeah, true. I mean they'll probably produce some answer except they're not going to be able to really produce millisecond by millisecond muscle control of how you stand up from your chair. But down to some level of abstraction where you can describe things by words, they might be able to give you a plan, but only under the condition that they've been trained to produce those kinds of plans. They're not going to be able to plan for situations where that they never encountered before. They basically are going to have to regurgitate the template that they've been trained on.

Lex Fridman
Just for the example of New York to Paris, is it going to start getting into trouble? Which layer of abstraction do you think you'll start? I can imagine almost every single part of that, an LLM would be able to answer somewhat accurately, especially when you're talking about New York and Paris, major cities.

Yann LeCun
I mean certainly LLM would be able to solve that problem if you fine tune it for it. And so I can't say that an LLM cannot do this, it can do this if you train it for it, there's no question down to a certain level where things can be formulated in terms of words. But if you want to go down to how you climb down the stairs or just stand up from your chair in terms of words, you can't do it. That's one of the reasons you need experience of the physical world, which is much higher bandwidth than what you can express in words, in human language.

Lex Fridman
So everything we've been talking about on the joint embedding space, is it possible that that's what we need for the interaction with physical reality on the robotics front, and then just the LLMs are the thing that sits on top of it for the bigger reasoning, about the fact that I need to book a plane ticket and I need to know how to go to the websites and so on.

Yann LeCun
Sure. And a lot of plans that people know about that are relatively high level are actually learned. Most people don't invent the plans by themselves. We have some ability to do this of course, obviously, but most plans that people use are plans that have been trained on, they've seen other people use those plans or they've been told how to do things, right? That you can't invent how you take a person who's never heard of airplanes and tell them how do you go from New York to Paris? And they're probably not going to be able to deconstruct the whole plan unless they've seen examples of that before. So certainly LLMs are going to be able to do this, but then how you link this from the low level of actions, that needs to be done with things like JEPA that basically lift the abstraction level of the representation without attempting to reconstruct the detail of the situation, that's why we need JEPAs for.

Lex Fridman
I would love to sort of linger on your skepticism around auto regressive LLMs. So one way I would like to test that skepticism is everything you say makes a lot of sense, but if I apply everything you said today and in general to I don't know, 10 years ago, maybe a little bit less, no, let's say three years ago, I wouldn't be able to predict the success of LLMs. So does it make sense to you that autoregressive LLMs are able to be so damn good?

Yann LeCun
Yes.

Lex Fridman
Can you explain your intuition? Because if I were to take your wisdom and intuition at face value, I would say there's no way autoregressive LLMs, one token at a time, would be able to do the kind of things they're doing.

Yann LeCun
No, there's one thing that autoregressive LLMs or that LLMs in general, not just the autoregressive one, but including the bird style bidirectional ones, are exploiting and its self-supervised learning, and I've been a very, very strong advocate of self-supervised learning for many years. So those things are a incredibly impressive demonstration that self-supervised learning actually works. The idea that started, it didn't start with BERT, but it was really kind of good demonstration with this. So the idea that you take a piece of text, you corrupt it, and then you train some gigantic neural net to reconstruct the parts that are missing. That has produced an enormous amount of benefits. It allowed us to create systems that understand language, systems that can translate hundreds of languages in any direction, systems that are multilingual, so it's a single system that can be trained to understand hundreds of languages and translate in any direction, and produce summaries and then answer questions and produce text. And then there's a special case of it, which is the auto regressive trick where you constrain the system to not elaborate a representation of the text from looking at the entire text, but only predicting a word from the words that

are come before. And you do this by constraining the architecture of the network, and that's what you can build an auto aggressive LLM from. So there was a surprise many years ago with what's called decoder only LLM. So since systems of this type that are just trying to produce words from the previous one and the fact that when you scale them up, they tend to really understand more about language. When you train them on lots of data, you make them really big. That was a surprise and that surprise occurred quite a while back, with work from Google, Meta, OpenAI, etc. - going back to the GPT kind of work, general pre-trained transformers.

Lex Fridman
You mean like GPT2? There's a certain place where you start to realize scaling might actually keep giving us an emergent benefit.

Yann LeCun
Yeah, I mean there were work from various places, but if you want to place it in the GPT timeline, that would be around GPT2, yeah.

Lex Fridman
Well, because you said it so charismatic and you said so many words, but self supervised learning, yes. But again, the same intuition you're applying to saying that auto aggressive LLMs cannot have a deep understanding of the world. If we just apply that, same intuition, does it make sense to you that they're able to form enough of a representation in the world to be damn convincing, essentially passing the original touring test with flying colors?

Yann LeCun
Well, we're fooled by their fluency, right? We just assume that if a system is fluent in manipulating language, then it has all the characteristics of human intelligence, but that impression is false. We're really fooled by it.

Lex Fridman
What do you think Alan Turing would say, without understanding anything, just hanging out with it?

Yann LeCun
Alan Turing would decide that a Turing test is a really bad test, okay? This is what the AI community has decided many years ago that the Turing test was a really bad test of intelligence.

Lex Fridman
What would Hans Marvek say about the larger language models?

Yann LeCun
Hans Marvek would say that Marvek Paradox still applies. Okay, we can pass -

Lex Fridman
You don't think he would be really impressed?

Yann LeCun
No, of course everybody would be impressed. But it's not a question of being impressed or not, it's the question of knowing what the limit of those systems can do. Again, they are impressive. They can do a lot of useful things. There's a whole industry that is being built around them. They're going to make progress, but there is a lot of things they cannot do, and we have to realize what they cannot do and then figure out how we get there. And I'm seeing this from basically 10 years of research on the idea of self-supervised learning, actually that's going back more than 10 years, but the idea of self-supervised learning. So basically capturing the internal structure of a piece of a set of inputs without training the system for any particular task, to learning representations. The conference I co-founded 14 years ago is called International Conference on Learning Representations. That's the entire issue that deep learning is dealing with, and it's been my obsession for almost 40 years now. So learning representation is really the thing. For the longest time, we could only do this with supervised learning, and then we started working on what we used to call unsupervised learning and revived the idea of unsupervised learning in the early 2000s with Yoshua Bengio and Geoffrey Hinton. Then discovered that supervised learning actually works pretty well if you can collect enough data. And so the whole idea of unsupervised, self-supervised learning kind of took a backseat for a bit, and then I tried to revive it in a big way starting in 2014, basically when we started FAIR and really pushing for finding new methods to do self-supervised learning both for text and for images and for video and audio. And some of that work has been incredibly successful. I mean, the reason why we have multilingual translation system, things to do, content moderation on Meta, for example, on Facebook, that are multilingual, that understand whether a piece of text is hate speech not or something, is due to that progress using self-supervised learning for NLP, combining this with transformer architectures and blah, blah, blah. But that's the big success of self-supervised learning. We had similar success in speech recognition, a system called WAVE2VEC, which is also a joint embedding architecture, by the way, trained with contrastive learning. And that system also can produce speech recognition systems that are multilingual with mostly unlabeled data and only need a few minutes of labeled data to actually do speech recognition, that's amazing. We have systems now based on those combination of ideas that can do real time translation of hundreds of languages into each other, speech to speech.

Lex Fridman
Speech to speech, even including, which is fascinating, languages that don't have written forms.

Yann LeCun
That’s right.

Lex Fridman
Just spoken only.

Yann LeCun
That’s right. We don’t go through text, it goes directly from speech to speech using an internal representation of speech units that are discrete, but it’s called Textless NLP. We used to call it this way. But yeah, so I mean incredible success there. And then for 10 years, we tried to apply this idea to learning representations of images by training a system to predict videos, learning intuitive physics by training a system to predict what’s going to happen in the video. And tried and tried and failed and failed, with generative models, with models that predict pixels. We could not get them to learn good representations of images. We could not get them to learn good representations of videos. And we tried many times, we published lots of papers on it, where they kind of sort of work, but not really great. They started working, we abandoned this idea of predicting every pixel and basically just doing the joint embedding and predicting and representation space, that works. So there’s ample evidence that we’re not going to be able to learn good representations of the real world using generative model. So I’m telling people, everybody’s talking about generative AI. If you’re really interested in human level AI, abandon the idea of generative AI.

Lex Fridman
Okay, but you really think it’s possible to get far with the joint embedding representation. So there’s common sense reasoning, and then there’s high level reasoning. I feel like those are two - the kind of reasoning that LLMs are able to do, okay, let me not use the word reasoning, but the kind of stuff that LLMs are able to do, seems fundamentally different than the common sense reasoning we use to navigate the world. It seems like we’re going to need both. Would you be able to get, with the joint embedding, which is JEPA type of approach, looking at video, would you be able to learn, let’s see, well, how to get from New York to Paris or how to understand the state of politics in the world today. These are things where various humans generate a lot of language and opinions on, in the space of language, but don’t visually represent that in any clearly compressible way.

Yann LeCun
Right. Well, there’s a lot of situations that might be difficult to, for a purely language based system to know. Okay, you can probably learn from reading texts, the entirety of the publicly available texts in the world that I cannot get from New York to Paris by snapping my fingers. That’s not going to work, right?

Lex Fridman
Yes.

Yann LeCun
But there's probably more complex scenarios of this type, which an LLM may never have encountered and may not be able to determine whether it's possible or not. So that link from the low level to the high level, the thing is that the high level that language expresses is based on the common experience of the low level, which LLMs currently do not have. When we talk to each other, we know we have a common experience of the world. A lot of it is similar, and LLMs don't have that.

Lex Fridman
But see, it's present. You and I have a common experience of the world in terms of the physics of how gravity works and stuff like this, and that common knowledge of the world, I feel like is there, in the language. We don't explicitly express it, but if you have a huge amount of text, you're going to get this stuff that's between the lines. In order to form a consistent world model, you're going to have to understand how gravity works, even if you don't have an explicit explanation of gravity. So even though in the case of gravity, there is explicit explanations of gravity in Wikipedia. But the stuff that we think of as common sense reasoning, I feel like to generate language correctly, you're going to have to figure that out. Now, you could say as you have, there's not enough text - sorry, okay. So, you don't think so?

Yann LeCun
No, I agree with what you just said, which is that to be able to do high level common sense, to have high level common sense, you need to have the low level common sense to build on top of.

Lex Fridman
But that's not there.

Yann LeCun
And that's not there in the LLMs. LLMs are purely trained from text. So then the other statement you made, I would not agree with, the fact that implicit in all languages in the world is the underlying reality, is a lot of underlying reality, which is not expressed in language.

Lex Fridman
Is that obvious to you?

Yann LeCun
Yeah, totally.

Lex Fridman
So all the conversations we had - okay, there's the dark web, meaning whatever, the private conversations like DMs and stuff like this, which is much, much larger probably than what's available, what LLMs are trained on.

Yann LeCun
You don't need to communicate the stuff that is common, right?

Lex Fridman
But the humor, all of it, no, you do, you don't need to, but it comes through. If I accidentally knock this over, you'll probably make fun of me in the content of the you making fun of me will be explanation of the fact that cups fall, and then gravity works in this way. And then you'll have some very vague information about what kind of things explode when they hit the ground. And then maybe you'll make a joke about entropy or something like this, then we'll never be able to reconstruct this again. You'll make a little joke like this and there'll be a trillion of other jokes. And from the jokes, you can piece together the fact that gravity works and mugs can break and all this kind of stuff. You don't need to see, it'll be very inefficient. It's easier to knock the thing over, but I feel like it would be there if you have enough of that data.

Yann LeCun
I just think that most of the information of this type that we have accumulated when we were babies, it's just not present in text, in any description, essentially.

Lex Fridman
And the sensory data is a much richer source for getting that kind of understanding.

Yann LeCun
I mean, there's 16,000 hours of wake time of a 4-year-old and tend to do 15 bites going through vision, just vision, there is a similar bandwidth of touch and a little less through audio. And then text, language doesn't come in until a year in life. And by the time you are nine years old, you've learned about gravity, you know about inertia, you know about gravity, the stability, you know about the distinction between animate and inanimate objects. You know by 18 months, you know about why people want to do things and you help them if they can't. I mean, there's a lot of things that you learn mostly by observation, really not even through interaction. In the first few months of life, babies don't really have any influence on the world, they can only observe. And you accumulate a gigantic amount of knowledge just from that. So that's what we're missing from current AI systems.

Lex Fridman
I think in one of your slides, you have this nice plot that is one of the ways you show that LLMs are limited. I wonder if you could talk about hallucinations from your perspectives, the

why hallucinations happen from large language models and to what degree is that a fundamental flaw of large language models?

Yann LeCun
Right, so because of the autoregressive prediction, every time an produces a token or a word, there is some level of probability for that word to take you out of the set of reasonable answers. And if you assume, which is a very strong assumption, that the probability of such error is that those errors are independent across a sequence of tokens being produced. What that means is that every time you produce a token, the probability that you stay within the set of correct answer decreases and it decreases exponentially.

Lex Fridman
So there's a strong, like you said, assumption there that if there's a non-zero probability of making a mistake, which there appears to be, then there's going to be a kind of drift.

Yann LeCun
Yeah, and that drift is exponential. It's like errors accumulate. So the probability that an answer would be nonsensical increases exponentially with the number of tokens.

Lex Fridman
Is that obvious to you, by the way? Well, mathematically speaking maybe, but isn't there a kind of gravitational pull towards the truth? Because on average, hopefully, the truth is well represented in the training set?

Yann LeCun
No, it's basically a struggle against the curse of dimensionality. So the way you can correct for this is that you fine tune the system by having it produce answers for all kinds of questions that people might come up with. Having it produce answers for all kinds of questions that people might come up with. And people are people, so a lot of the questions that they have are very similar to each other, so you can probably cover 80% or whatever of questions that people will ask by collecting data and then you fine tune the system to produce good answers for all of those things, and it's probably going to be able to learn that because it's got a lot of capacity to learn. But then there is the enormous set of prompts that you have not covered during training, and that set is enormous, like within the set of all possible prompts, the proportion of prompts that have been used for training is absolutely tiny, it's a tiny, tiny, tiny subset of all possible prompts. And so the system will behave properly on the prompts that has been either trained, pre-trained, or fine-tuned, but then there is an entire space of things that it cannot possibly have been trained on because the number is gigantic. So whatever training the system has been subject to produce appropriate answers, you can break it by finding out a prompt that will be outside of the set of prompts that's been trained on, or things that are similar, and then it will just spew complete nonsense.

Lex Fridman
When you say prompt, do you mean that exact prompt or do you mean a prompt that's in many parts, very different than? Is it that easy to ask a question or to say a thing that hasn't been said before on the internet?

Yann LeCun
People have come up with things where you put essentially a random sequence of characters in the prompt and that's enough to throw the system into a mode where it is going to answer something completely different than it would have answered without this. So that's a way to jailbreak the system, basically go outside of its conditioning.

Lex Fridman
That's a very clear demonstration of it, but of course, that goes outside of what is designed to do, right? If you actually stitch together reasonably grammatical sentences, is it that easy to break it?

Yann LeCun
Yeah, some people have done things like, you write a sentence in English or you ask a question in English and it produces a perfectly fine answer and then you just substitute a few words by the same word in another language and all of a sudden the answer is complete nonsense.

Lex Fridman
What I'm saying is, which fraction of prompts that humans are likely to generate are going to break the system?

Yann LeCun
The problem is that there is a long tail, this is an issue that a lot of people have realized in social networks and stuff like that, which is there's a very, very long tail of things that people will ask and you can fine tune the system for the 80% or whatever of the things that most people will ask. And then this long tail is so large that you're not going to be able to fine tune the system for all the conditions. And in the end, the system ends up being a giant lookup table essentially, which is not really what you want, you want systems that can reason, certainly that can plan. The type of reasoning that takes place in LLM is very, very primitive, and the reason you can tell is primitive is because the amount of computation that is spent per token produced is constant. So if you ask a question and that question has an answer in a given number of token, the amount of computation devoted to computing that answer can be exactly estimated. It's the size of the prediction network with its 36 layers or 92 layers or whatever it is multiply by number of tokens, that's it. And so essentially, it doesn't matter if the question being asked is simple to answer, complicated to answer, impossible to answer because it's a decidable or something, the amount of computation the system will be able to devote to the answer is constant or is proportional to number of token produced in the

answer. This is not the way we work, the way we reason is that when we're faced with a complex problem or a complex question, we spend more time trying to solve it and answer it because it's more difficult.

Lex Fridman
There's a prediction element, there's an iterative element where you're adjusting your understanding of a thing by going over and over and over, there's a hierarchical elements on. Does this mean it's a fundamental flaw of LLMs or does it mean that -

Yann LeCun
Yeah.

Lex Fridman
- there's more part to that question, now you're just behaving like an LLM, immediately answering. No, that it's just the low level world model on top of which we can then build some of these kinds of mechanisms, like you said, persistent long-term memory or reasoning, so on. But we need that world model that comes from language. Maybe it is not so difficult to build this kind of reasoning system on top of a well constructed world model.

Yann LeCun
Whether it's difficult or not, the near future will say because a lot of people are working on reasoning and planning abilities for dialogue systems. Even if we restrict ourselves to language, just having the ability to plan your answer before you answer in terms that are not necessarily linked with the language you're going to use to produce the answer, so this idea of this mental model that allows you to plan what you're going to say before you say it, that is very important. I think there's going to be a lot of systems over the next few years that are going to have this capability, but the blueprint of those systems will be extremely different from auto aggressive LLMs. It's the same difference as the difference between what psychologists call system one and system two in humans, so system one is the type of task that you can accomplish without deliberately consciously think about how you do them, you just do them, you've done them enough that you can just do it subconsciously without thinking about them. If you're an experienced driver, you can drive without really thinking about it and you can talk to someone at the same time or listen to the radio. If you are a very experienced chess player, you can play against a non- experienced chess player without really thinking either, you just recognize the pattern and you play. That's system one, so all the things that you do instinctively without really having to deliberately plan and think about it. And then there is all the tasks where you need to plan, so if you are a not too experienced chess player or you are experienced where you play against another experienced chess player, you think about all kinds of options, you think about it for a while and you are much better if you have time to think about it than you are if you play blitz with limited time. So this type of deliberate planning, which uses your internal world model, that's system two, this is what LLMs currently cannot do. How do we get them to do this? How do we build a

system that can do this kind of planning or reasoning that devotes more resources to complex problems than to simple problems? And it's not going to be a regressive prediction of tokens, it's going to be more something akin to inference of little variables in what used to be called probabilistic models or graphical models and things of that type. Basically, the principle is like this, the prompt is like observed variables, and what the model does, is that basically, it can measure to what extent an answer is a good answer for a prompt. So think of it as some gigantic neural net, but it's got only one output, and that output is a scaler number, which is, let's say, zero, if the answer is a good answer for the question and a large number, if the answer is not a good answer for the question. Imagine you had this model, if you had such a model, you could use it to produce good answers, the way you would do is, produce the prompt and then search through the space of possible answers for one that minimizes that number, that's called an energy based model.

Lex Fridman
But that energy based model would need the model constructed by the LLM?

Yann LeCun
Well, so really what you need to do would be to not search over possible strings of text that minimize that energy. But what you would do, we do this in abstract representation space, so in the space of abstract thoughts, you would elaborate a thought using this process of minimizing the output of your model, which is just a scaler, it's an optimization process. So now the way the system produces its sensor is through optimization by minimizing an objective function basically. And we're talking about inference, we're not talking about training, the system has been trained already. Now we have an abstract representation of the thought of the answer, representation of the answer, we feed that to basically an autoregressive decoder, which can be very simple, that turns this into a text that expresses this thought. So that, in my opinion, is the blueprint of future data systems, they will think about their answer, plan their answer by optimization before turning it into text, and that is turning complete.

Lex Fridman
Can you explain exactly what the optimization problem there is? What's the objective function? Just linger on it, you briefly described it, but over what space are you optimizing?

Yann LeCun
The space of representations.

Lex Fridman
It goes abstract representation?

Yann LeCun
You have an abstract representation inside the system, you have a prompt, the prompt goes through an encoder, produces a representation, perhaps goes through a predictor that predicts a representation of the proper answer. But that representation may not be a good answer because there might be some complicated reasoning you need to do, so then you have another process that takes the representation of the answers and modifies it so as to minimize a cost function that measures to what extent the answer is a good answer for the question. Now we ignore the issue for a moment of how you train that system to measure whether an answer is a good answer for a fraction.

Lex Fridman
Sure. Suppose such a system could be created, but what's this search like process?

Yann LeCun
It's an optimization process. You can do this if the entire system is differentiable, that scaler output is the result of learning the representation of the answers to some neural net. Then by gradient descent, by back propagating gradients, you can figure out how to modify the representation of the answers so as to minimize that.

Lex Fridman
That's still a gradient based?

Yann LeCun
It's gradient based inference. So now you have a representation of the answer in abstract space, now you can turn it into text. And the cool thing about this is that the representation now can be optimized through gradient descent, but also is independent of the language in which you're going to express the answer.

Lex Fridman
Right. So you're operating in the subtract representation. This goes back to the joint embedding, that it's better to work in the space of, I don't know, or to romanticize the notion like space of concepts versus the space of concrete sensory information.

Yann LeCun
Right.

Lex Fridman
But can this do something like reasoning, which is what we're talking about?

Yann LeCun
Well, not really, only in a very simple way. Basically, you can think of those things as doing the optimization I was talking about, except they optimize in the discrete space, which is

the space of possible sequences of tokens. And they do this optimization in a horribly inefficient way, which is generate a lot of hypothesis and then select the best ones. And that's incredibly wasteful in terms of competition because you basically have to run your LLM for every possible generative sequence and it's incredibly wasteful. So it's much better to do an optimization in continuous space where you can do gradient and descent as opposed to generate tons of things and then select the best, you just iteratively refine your answer to go towards the best, that's much more efficient. But you can only do this in continuous spaces with differentiable functions.

Lex Fridman
You're talking about the ability to think deeply or to reason deeply, how do you know what is an answer that's better or worse based on deep reasoning?

Yann LeCun
Then we are asking the question of, conceptually, how do you train an energy based model? Energy based model is a function with a scaler output, just a number, you give it two inputs, X and Y, and it tells you whether Y is compatible with X or not. X, you observe, let's say it's a prompt, an image, a video, whatever, and Y is a proposal for an answer, a continuation of video, whatever and it tells you whether Y is compatible with X. And the way it tells you that Y is compatible with X is that the output of that function would be zero if Y is compatible with X and would be a positive number, non-zero, if Y is not compatible with X. How do you train a system like this at a completely general level, is you show it pairs of X and Ys that are compatible, a question and the corresponding answer, and you train the parameters of the big neural net inside to produce zero. Now that doesn't completely work because the system might decide, well, I'm just going to say zero for everything, so now you have to have a process to make sure that for a wrong Y, the energy would be larger than zero. And there you have two options, one is contrastive method, so contrastive method is, you show an X and a bad Y and you tell the system, well, give a high energy to this, push up the energy, change the weights in the neural net that confuse the energy so that it goes up. So that's contrasting methods. The problem with this is, if the space of Y is large, the number of such contrasting samples are going to have to show is gigantic. But people do this, they do this when you train a system with RLHF, basically what you're training is what's called a reward model, which is basically an objective function that tells you whether an answer is good or bad, and that's basically exactly what this is. So we already do this to some extent, we're just not using it for inference, we're just using it for training. There is another set of methods which are non-contrastive, and I prefer those, and those non-contrastive methods basically say, the energy function needs to have low energy on pairs of XYs that are compatible that come from your training set. How do you make sure that the energy is going to be higher everywhere else? And the way you do this is by having a regularizer, a criterion, a term in your cost function that basically minimizes the volume of space that can take low energy. And the precise way to do this is all kinds of different specific ways to do this depending on the architecture, but that's the basic principle. So that if you push down the energy function

for particular regions in the XY space, it will automatically go up in other places because there's only a limited volume of space that can take low energy by the construction of the system or by the regularizing function.

Lex Fridman
We've been talking very generally, but what is a good X and a good Y? What is a good representation of X and Y? Because we've been talking about language and if you just take language directly that presumably is not good, so there has to be some kind of abstract representation of ideas.

Yann LeCun
You can do this with language directly by just, X is a text and Y is a continuation of that text.

Lex Fridman
Yes.

Yann LeCun
Or X is a question, Y is the answer.

Lex Fridman
But you're saying that's not going to take it, that's going to do what LLMs are doing.

Yann LeCun
Well, no, it depends on how the internal structure of the system is built. If the internal structure of the system is built in such a way that inside of the system there is a latent variable, let's call it Z, that you can manipulate so as to minimize the output energy, then that Z can be viewed as a representation of a good answer that you can translate into a Y that is a good answer.

Lex Fridman
This system could be trained in a very similar way?

Yann LeCun
Very similar way, but you have to have this way preventing collapse of ensuring that there is high energy for things you don't train it on. And currently, it's very implicit in LLM, it's done in a way that people don't realize it's being done, but it is being done. It is due to the fact that when you give a high probability to a word, automatically, you give low probability to other words because you only have a finite amount of probability to go around right there to sum to one. So when you minimize the cross entropy or whatever, when you train your LLM to predict the next word, you are increasing the probability your system will give to the correct word, but you're also decreasing the probability it will give to the incorrect words. Now, indirectly, that gives a high probability to sequences of words that are good and low

probability to sequences of words that are bad, but it's very indirect. And it's not obvious why this actually works at all because you're not doing it on the joint probability of all the symbols in a sequence, you factorize that probability in terms of conditional probabilities over successive tokens.

Lex Fridman
How do you do this for visual data?

Yann LeCun
We've been doing this with I-JEPA architectures, basically-

Lex Fridman
The joint embedding.

Yann LeCun
- I-JEPA. So there the compatibility between two things is, here's an image or a video, here is a corrupted, shifted or transformed version of that image or video or masked. And then the energy of the system is the prediction error of the predicted representation of the good thing versus the actual representation of the good thing. So you run the corrupted image to the system, predict the representation of the good input uncorrupted, and then compute the prediction error, that's the energy of the system. So this system will tell you if this is a good image and this is a corrupted version, it will give you zero energy if those two things, effectively, one of them is a corrupted version of the other, it gives you a high energy if the two images are completely different.

Lex Fridman
And hopefully that whole process gives you a really nice compressed representation of a visual reality?

Yann LeCun
And we know it does because then we use those representations as input to a classification system or something and that it works.

Lex Fridman
And then that classification system works really nicely, okay. Well, so to summarize, you recommend in a spicy way that only Yann LeCun can, you recommend that we abandon generative models in favor of joint embedding architectures?

Yann LeCun
Yes.

Lex Fridman
Abandon autoregressive generation.

Yann LeCun
Yes.

Lex Fridman
This feels like court testimony, abandon probabilistic models in favor of energy based models as we talked about, abandon contrastive methods in favor of regularized methods. And let me ask you about this, you've been for a while, a critic of reinforcement learning.

Yann LeCun
Yes.

Lex Fridman
The last recommendation is that we abandon RL in favor of model predictive control, as you were talking about, and only use RL when planning doesn't yield the predicted outcome, and we use RL in that case to adjust the world model or the critic.

Yann LeCun
Yes.

Lex Fridman
You've mentioned RLHF, reinforcement learning with human feedback, why do you still hate reinforcement learning?

Yann LeCun
I don't hate reinforcement learning, and I think-

Lex Fridman
It's all love, yes.

Yann LeCun
- I think it should not be abandoned completely, but I think it's use should be minimized because it's incredibly inefficient in terms of samples. And so the proper way to train a system is to first have it learn good representations of the world and world models from mostly observation, maybe a little bit of interactions.

Lex Fridman
And then steered based on that, if the representation is good, then the adjustments should be minimal.

Yann LeCun
Yeah. Now there's two things, if you've learned a world model, you can use the world model to plan a sequence of actions to arrive at a particular objective, you don't need RL unless the way you measure whether you succeed might be in exact. Your idea of whether you are going to fall from your bike might be wrong, or whether the person you're fighting with MMA who's going to do something and they do something else. So there's two ways you can be wrong, either your objective function does not reflect the actual objective function you want to optimize or your world model is inaccurate, so the prediction you were making about what was going to happen in the world is inaccurate. If you want to adjust your world model while you are operating in the world or your objective function, that is basically in the realm of RL, this is what RL deals with to some extent, so adjust your word model. And the way to adjust your word model even in advance is to explore parts of the space where you know that your world model is inaccurate, that's called curiosity basically, or play. When you play, you explore parts of the space that you don't want to do for real because it might be dangerous, but you can adjust your world model without killing yourself basically. So that's what you want to use RL for, when it comes time to learning a particular task, you already have all the good representations, you already have your world model, but you need to adjust it for the situation at hand, that's when you use RL.

Lex Fridman
Why do you think RLHF works so well? This enforcement learning with human feedback, why did it have such a transformational effect on large language models than before?

Yann LeCun
What's had the transformational effect is human feedback, there is many ways to use it, and some of it is just purely supervised, actually, it's not really reinforcement learning.

Lex Fridman
It's the HF?

Yann LeCun
It's the HF, and then there is various ways to use human feedback. So you can ask humans to rate multiple answers that are produced by world model, and then what you do is you train an objective function to predict that rating, and then you can use that objective function to predict whether an answer is good and you can back propagate gradient to this to fine tune your system so that it only produces highly rated answers. That's one way, so in RL, that means training what's called a reward model, so something that basically is a small neural net that estimates to what extent an answer is good. It's very similar to the objective I was talking about earlier for planning, except now it's not used for planning, it's used for fine-tuning your system. I think it would be much more efficient to use it for planning, but currently, it's used to fine tune the parameters of the system. There's several ways to do this, some of them are supervised, you just ask a human person like, what is a good answer

for this? Then you just type the answer. There's lots of ways that those systems are being adjusted.

Lex Fridman
Now, a lot of people have been very critical of the recently released Google's Gemini 1.5 for essentially, in my words, I could say super woke in the negative connotation of that word. There is some almost hilariously absurd things that it does, like it modifies history like generating images of a black George Washington, or perhaps more seriously something that you commented on Twitter, which is refusing to comment on or generate images or even descriptions of Tiananmen Square or The Tank Man, one of the most legendary protest images in history. Of course, these images are highly censored by the Chinese government and therefore, everybody started asking questions of what is the process of designing these LLMs? What is the role of censorship and all that kind of stuff? So you commented on Twitter saying that open source is the answer.

Yann LeCun
Yeah.

Lex Fridman
Essentially, so can you explain?

Yann LeCun
I actually made that comment on just about every social network I can, and I've made that point multiple times in various forums. Here's my point of view on this, people can complain that AI systems are biased and they generally are biased by the distribution of the training data that they've been trained on that reflects biases in society, and that is potentially offensive to some people or potentially not. And some techniques to de-bias then become offensive to some people because of historical incorrectness and things like that. And so you can ask two questions, the first question is, is it possible to produce an AI system that is not biased? And the answer is, absolutely not. And it's not because of technological challenges, although they are technological challenges to that, it's because bias is in the eye of the beholder. Different people may have different ideas about what constitutes bias for a lot of things, there are facts that are indisputable, but there are a lot of opinions or things that can be expressed in different ways. And so you cannot have an unbiased system, that's just an impossibility. And so what's the answer to this? And the answer is the same answer that we found in liberal democracy about the press, the press needs to be free and diverse. We have free speech for a good reason, is because we don't want all of our information to come from a unique source because that's opposite to the whole idea of democracy and progressive ideas and even science. In science, people have to argue for different opinions and science makes progress when people disagree and they come up with an answer and consensus forms, and it's true in all democracies around the world. There is a future which is already happening where every single one of our interaction with the digital world will be

mediated by AI systems, AI assistance. We're going to have smart glasses, you can already buy them from Meta, the Ray-Ban Meta where you can talk to them and they are connected with an LLM and you can get answers on any question you have. Or you can be looking at a monument and there is a camera in the glasses you can ask it like, what can you tell me about this building or this monument? You can be looking at a menu in a foreign language, and I think we will translate it for you, or we can do real time translation if we speak different languages. So a lot of our interactions with the digital world are going to be mediated by those systems in the near future. Increasingly, the search engines that we're going to use are not going to be search engines, they're going to be dialogue systems that we just ask a question and it will answer and then point you to perhaps appropriate reference for it. But here is the thing, we cannot afford those systems to come from a handful of companies on the west coast of the US because those systems will constitute the repository of all human knowledge, and we cannot have that be controlled by a small number of people. It has to be diverse for the same reason the press has to be diverse, so how do we get a diverse set of AI assistance? It's very expensive and difficult to train a base model, a base LLM at the moment, in the future it might be something different, but at the moment, that's an LLM. So only a few companies can do this properly. And if some of those top systems are open source, anybody can use them, anybody can fine tune them. If we put in place some systems that allows any group of people, whether they are individual citizens, groups of citizens, government organizations, NGOs, companies, whatever, to take those open source AI systems and fine tune them for their own purpose on their own data, then we're going to have a very large diversity of different AI systems that are specialized for all of those things. I tell you, I talked to the French government quite a bit, and the French government will not accept that the digital diet of all their citizens be controlled by three companies on the west coast of the US. That's just not acceptable, it's a danger to democracy regardless of how well-intentioned those companies are, and it's also a danger to local culture, to values, to language. I was talking with the founder of Infosys in India, he's funding a project to fine tune Llama 2, the open source model produced by Meta, so that Llama 2 two speaks all 22 official languages in India, it is very important for people in India. I was talking to a former colleague of mine, Moustapha Cisse, who used to be a scientist at Fair and then moved back to Africa, created a research lab for Google in Africa and now has a new startup Co-Kera. And what he's trying to do, is basically have LLM that speak the local languages in Senegal so that people can have access to medical information because they don't have access to doctors, it's a very small number of doctors per capita in Senegal. You can't have any of this unless you have open source platforms, so with open source platforms, you can have AI systems that are not only diverse in terms of political opinions or things of that - AI systems that are not only diverse in terms of political opinions or things of that type, but in terms of language, culture, value systems, political opinions, technical abilities in various domains, and you can have an industry, an ecosystem of companies that fine tune those open source systems for vertical applications in industry. I don't know, a publisher has thousands of books and they want to build a system that allows a customer to just ask a question about the content of any of their books, you need to train on their proprietary data. You have a company, we have one within Meta, it's called Metamate, and it's basically an LLM that can answer any question about internal stuff about the company, very useful. A lot of companies want this. A lot of companies want this not just for their employees, but also for their customers, to take care of their customers. So the only way you're going to have an AI industry, the only way you're going to have AI systems that are not uniquely biased is if you have open source platforms on top of which any group can build specialized systems. So the direction of inevitable direction of history is that the vast majority of AI systems will be built on top of open source platforms.

Lex Fridman
So that's a beautiful vision. So meaning a company like Meta or Google or so on should take only minimal fine-tuning steps after building the foundation pre-trained model as few steps as possible.

Yann LeCun
Basically.

Lex Fridman
Can Meta afford to do that?

Yann LeCun
No.

Lex Fridman
So I don't know if you know this, but companies are supposed to make money somehow and open source is giving away I don't know. Mark made a video, Mark Zuckerberg, very sexy video talking about 350,000 Nvidia H100s.

Yann LeCun
Yeah, it's actually -

Lex Fridman
The math of that is just for the GPUs, that's 100 billion plus the infrastructure for training everything. So I'm no business guy, but how do you make money on that? So the division you paint is a really powerful one, but how is it possible to make money?

Yann LeCun
Okay, so you have several business models, right?

Lex Fridman
Mmhmm.

Yann LeCun
The business model that Meta is built around is you offer a service and the financing of that service is either through ads or through business customers. So for example, if you have an LLM that can help a mom-and-pop pizza place by talking to the customers through WhatsApp, and so the customers can just order a pizza and the system will just ask them, "What topping do you want or what size, blah, blah, blah." The business will pay for that, okay? That's a model. Otherwise, if it's a system that is on the more classical services, it can be ad supported or there's several models. But the point is, if you have a big enough potential customer base and you need to build that system anyway for them, it doesn't hurt you to actually distribute it to the open source.

Lex Fridman
Again, I'm no business guy, but if you release the open source model, then other people can do the same kind of task and compete on it, basically provide fine-tuned models for businesses.

Yann LeCun
Sure.

Lex Fridman
By the way, I'm a huge fan of all this, but is the bet that Meta is making, it's like, "We'll do a better job of it?"

Yann LeCun
Well, no. The bet is more, "We already have a huge user base and customer base-

Lex Fridman
Ah, right.

Yann LeCun
- so it's going to be useful to them. Whatever we offer them is going to be useful and there is a way to derive revenue from this.

Lex Fridman
Sure.

Yann LeCun
It doesn't hurt that we provide that system or the base model, the foundation model in open source for others to build applications on top of it too. If those applications turn out to be useful for our customers, we can just buy it from them. It could be that they will improve the platform. In fact, we see this already. There is literally millions of downloads of Llama 2 and thousands of people who have provided ideas about how to make it better. So this clearly

accelerates progress to make the system available to a wide community of people, and there's literally thousands of businesses who are building applications with it. So Meta's ability to derive revenue from this technology is not impaired by the distribution of base models in open source.

Lex Fridman
The fundamental criticism that Gemini is getting is that as you point out on the West Coast, just to clarify, we're currently on the East Coast where I would suppose Meta AI headquarters would be. So there are strong words about the West Coast, but I guess the issue that happens is I think it's fair to say that most tech people have a political affiliation with the left wing. They lean left. So the problem that people are criticizing Gemini with is that there's in that de-biasing process that you mentioned, that their ideological lean becomes obvious. Is this something that could be escaped? You're saying open source is the only way.

Yann LeCun
Yes.

Lex Fridman
Have you witnessed this kind of ideological lean that makes engineering difficult?

Yann LeCun
No, I don't think the issue has to do with the political leaning of the people designing those systems. It has to do with the acceptability or political leanings of their customer base or audience. So a big company cannot afford to offend too many people, so they're going to make sure that whatever product they put out is safe, whatever that means. It's very possible to overdo it, and it's impossible to do it properly for everyone. You're not going to satisfy everyone. So that's what I said before, you cannot have a system that is perceived as unbiased by everyone. It's going to be you push it in one way, one set of people are going to see it as biased, and then you push it the other way and another set of people is going to see it as biased. Then in addition to this, there's the issue of if you push the system perhaps a little too far in one direction, it's going to be non-factual. You're going to have Black Nazi soldiers in uniform.

Lex Fridman
Yeah, we so we should mention image generation of Black Nazi soldiers, which is not factually accurate.

Yann LeCun
Right, and can be offensive for some people as well. So it's going to be impossible to produce systems that are unbiased for everyone. So the only solution that I see is diversity.

Lex Fridman
Diversity in the full meaning of that word, diversity of in every possible way.

Yann LeCun
Yeah.

Lex Fridman
Marc Andreessen just tweeted today. Let me do a TL;DR. The conclusion is only startups and open source can avoid the issue that he's highlighting with big tech. He's asking, "Can Big Tech actually field generative AI products?" (1) Ever-escalating demands from internal activists, employee mobs, crazed executives, broken boards, pressure groups, extremist regulators, government agencies, the press, in quotes, "experts" and everything corrupting the output. (2) Constant risk of generating a bad answer or drawing a bad picture or rendering a bad video who knows what is going to say or do at any moment. (3) Legal exposure, product liability, slander, election law, many other things and so on, anything that makes Congress mad. (4) Continuous attempts to tighten grip on acceptable output, degrade the model, how good it actually is, in terms of usable and pleasant to use and effective and all that kind of stuff. (5) Publicity of bad text, images, video actual puts those examples into the training data for the next version and so on. So he just highlights how difficult this is from all kinds of people being unhappy. He said you can't create a system that makes everybody happy.

Yann LeCun
Yes.

Lex Fridman
So if you're going to do the fine-tuning yourself and keep it close source, essentially, the problem there is then trying to minimize the number of people who are going to be unhappy.

Yann LeCun
Yep.

Lex Fridman
You're saying that almost impossible to do, and there are better ways to do open source

Yann LeCun
Basically. Yeah. Mark is right about a number of things that you list that indeed scare large companies. Certainly, congressional investigations is one of them, legal liability, making things that get people to hurt themselves or hurt others. Big companies are really careful about not producing things of this type because they don't want to hurt anyone, first of all, and then second, they want to preserve their business. So it's essentially impossible for systems like this that can inevitably formulate political opinions, and opinions about various

things that may be political or not, but that people may disagree about, about moral issues and questions about religion and things like that or cultural issues that people from different communities would disagree with in the first place. So there's only a relatively small number of things that people will agree on are basic principles, but beyond that, if you want those systems to be useful, they will necessarily have to offend a number of people inevitably.

Lex Fridman
So open source is just better and then you get-

Yann LeCun
Diversity is better, right?

Lex Fridman
And open source enables diversity.

Yann LeCun
That's right. Open source enables diversity.

Lex Fridman
This can be a fascinating world where if it's true that the open source world, if Meta leads the way and creates this open source foundation model world, governments will have a fine-tuned model and then potentially, people that vote left and right will have their own model and preference to be able to choose and it will potentially divide us even more. But that's on us humans. We get to figure out basically the technology enables humans to human more effectively, and all the difficult ethical questions that humans raise will just leave it up to us to figure that out.

Yann LeCun
Yeah, there are some limits. The same way there are limits to free speech. There has to be some limit to the kind of stuff that those systems might be authorized to produce, some guardrails. So that's one thing I'd be interested in, which is in the type of architecture that we were discussing before where the output of the system is a result of an inference to satisfy an objective, that objective can include guardrails, and we can put guardrails in open source systems. If we eventually have systems that are built with this blueprint, we can put guardrails in those systems that guarantee that there is a minimum set of guardrails that make the system non-dangerous and non-toxic, etc. - basic things that everybody would agree on. Then the fine-tuning that people will add or the additional guardrails that people will add will cater to their community, whatever it is.

Lex Fridman

The fine-tuning will be more about the gray areas of what is hate speech, what is dangerous and all that kind of stuff, but it's the-

Yann LeCun

Or different value systems.

Lex Fridman

Still value systems. But still even with the objectives of how to build a bioweapon, for example, I think something you've commented on, or at least there's a paper where a collection of researchers is trying to understand the social impacts of these LLMs. I guess one threshold that's nice is, does the LLM make it any easier than a search would, like a Google search would?

Yann LeCun

Right. So the increasing number of studies on this seems to point to the fact that it doesn't help. So having an LLM doesn't help you design or build a bioweapon or a chemical weapon if you already have access to a search engine and their library. So the increased information you get or the ease with which you get it doesn't really help you. That's the first thing. The second thing is, it's one thing to have a list of instructions of how to make a chemical weapon, for example, a bioweapon. It's another thing to actually build it, and it's much harder than you might think, and then LLM will not help you with that. In fact, nobody in the world, not even countries used bioweapons because most of the time they have no idea how to protect their own populations against it. So it's too dangerous, actually, to ever use, and it's, in fact, banned by international treaties. Chemical weapons is different. It's also banned by treaties, but it's the same problem. It's difficult to use in situations that doesn't turn against the perpetrators, but we could ask Elon Musk. I can give you a very precise list of instructions of how you build a rocket engine. Even if you have a team of 50 engineers that are really experienced building it, you're still going to have to blow up a dozen of them before you get one that works. It's the same with chemical weapons or bioweapons or things like this, it requires expertise in the real world that the LLM is not going to help you with.

Lex Fridman

It requires even the common sense expertise that we've been talking about, which is how to take language-based instructions and materialize them in the physical world requires a lot of knowledge that's not in the instructions.

Yann LeCun

Yeah, exactly. A lot of biologists have posted on this actually, in response to those things saying, "Do you realize how hard it is to actually do the lab work?" Like, "No, this is not trivial."

Lex Fridman
Yeah, and Hans Moravec comes to light once again. Just to linger on Llama, Marc announced that Llama 3 is coming out eventually. I don't think there's a release date, but what are you most excited about? First of all, Llama 2 that's already out there and maybe the future a Llama 3, 4, 5, 6, 10, just the future of the open source under Meta?

Yann LeCun
Well, a number of things. So there's going to be various versions of Llama that are improvements of previous Llamas, bigger, better, multimodal, things like that. Then in future generations, systems that are capable of planning that really understand how the world works, maybe are trained from video, so they have some world model maybe capable of the type of reasoning and planning I was talking about earlier. How long is that going to take? When is the research that is going in that direction going to feed into the product line if you want of Llama? I don't know. I can't tell you. There's a few breakthroughs that we have to basically go through before we can get there, but you'll be able to monitor our progress because we publish our research. So last week we published the V-JEPA work, which is a first step towards training systems for video. Then the next step is going to be world models based on this type of idea training from video. There's similar work at DeepMind also and taking place people, and also at UC Berkeley on world models and video. A lot of people are working on this. I think a lot of good ideas are appearing. My bet is that those systems are going to be JEPA light, they're not going to be generative models, and we'll see what the future will tell. There's really good work, a gentleman called Danijar Hafner who is now DeepMind, who's worked on models of this type that learn representations and then use them for planning or learning tasks by reinforcement training and a lot of work at Berkeley by Pieter Abbeel, Sergey Levine, a bunch of other people of that type I'm collaborating with actually in the context of some grants with my NYU hat. Then collaboration is also through Meta 'cause the lab at Berkeley is associated with Meta in some way, so with fair. So I think it is very exciting. I haven't been that excited about the direction of machine learning and AI since 10 years ago when Fairway was started. Before that, 30 years ago, we were working, oh, sorry, 35 on combination nets and the early days of neural nets. So I'm super excited because I see a path towards potentially human-level intelligence with systems that can understand the world, remember, plan, reason. There is some set of ideas to make progress there that might have a chance of working, and I'm really excited about this. What I like is that somewhat we get on to a good direction and perhaps succeed before my brain turns to a white sauce or before I need to retire.

Lex Fridman
Yeah. Yeah. Is it beautiful to you just the amount of GPUs involved, the whole training process on this much compute, just zooming out, just looking at earth and humans together have built these computing devices and are able to train this one brain, then we then open source, like giving birth to this open source brain trained on this gigantic compute system, there's just the details of how to train on that, how to build the infrastructure and the

hardware, the cooling, all of this kind of stuff, or are you just still that most of your excitement is in the theory aspect of it, meaning the software?

Yann LeCun
I used to be a hardware guy many years ago.

Lex Fridman
Yes. Yes, that's right.

Yann LeCun
Decades ago.

Lex Fridman
Hardware has improved a little bit. Changed-

Yann LeCun
A little bit.

Lex Fridman
- a little bit, yeah.

Yann LeCun
Certainly, scale is necessary but not sufficient.

Lex Fridman
Absolutely.

Yann LeCun
So we certainly need competition. We're still far in terms of compute power from what we would need to match the compute power of the human brain. This may occur in the next couple of decades, but we're still some ways away. Certainly, in terms of power efficiency, we're really far, so there's a lot of progress to make in hardware. Right now, a lot of the progress is, there's a bit coming from silicon technology, but a lot of it coming from architectural innovation and quite a bit coming from more efficient ways of implementing the architectures that have become popular, basically combination of transformers and com nets, and so there's still some ways to go until we are going to saturate. We're going to have to come up with new principles, new fabrication technology, new basic components - perhaps based on different principles. I know their's classical, digital -

Lex Fridman
Interesting. So you think in order to build AMI, we potentially might need some hardware innovation too.

Yann LeCun
Well, if we want to make it ubiquitous, yeah, certainly, 'cause we're going to have to reduce the - you know - compute - power consumption. A GPU today is half a kilowatt to a kilowatt. Human brain is about 25 watts, and a GPU is way below the power of the human brain. You need something like 100,000 or a million to match it, so we are off by a huge factor here.

Lex Fridman
You often say that a GI is not coming soon, meaning not this year, not the next few years, potentially farther away. What's your basic intuition behind that?

Yann LeCun
So first of all, it's not going to be an event. The idea somehow, which is popularized by science fiction and Hollywood, that somehow somebody is going to discover the secret to AGI or human-level AI or AMI, whatever you want to call it, and then turn on a machine and then we have AGI, that's just not going to happen. It's not going to be an event. It's going to be gradual progress. Are we going to have systems that can learn from video how the world works and learn good representations? Yeah. Before we get them to the scale and performance that we observe in humans it's going to take quite a while. It's not going to happen in one day. Are we going to get systems that can have large amount of associated memory so they can remember stuff? Yeah, but same, it's not going to happen tomorrow. There is some basic techniques that need to be developed. We have a lot of them, but to get this to work together with a full system is another story. Are we going to have systems that can reason and plan perhaps along the lines of objective-driven AI architectures that I described before? Yeah, but before we get this to work properly, it's going to take a while. Before we get all those things to work together, and then on top of this, have systems that can learn hierarchical planning, hierarchical representations, systems that can be configured for a lot of different situation at hand, the way the human brain can, all of this is going to take at least a decade and probably much more because there are a lot of problems that we're not seeing right now that we have not encountered, so we don't know if there is an easy solution within this framework. So it's not just around the corner. I've been hearing people for the last 12, 15 years claiming that AGI is just around the corner and being systematically wrong. I knew they were wrong when they were saying it. I called their bullshit.

Lex Fridman
First of all, from the birth of the term artificial intelligence, there has been a eternal optimism that's perhaps unlike other technologies. Is it a Moravec's paradox, the explanation for why people are so optimistic about AGI?

Yann LeCun
Don't think it's just Moravec's paradox. Moravec's paradox is a consequence of realizing that the world is not as easy as we think. So first of all, intelligence is not a linear thing that you

can measure with a scale or with a single number. Can you say that humans are smarter than orangutans? In some ways, yes, but in some ways, orangutans are smarter than humans in a lot of domains that allows them to survive in the forest, for example.

Lex Fridman
So IQ is a very limited measure of intelligence. Human intelligence is bigger than what IQ, for example, measures.

Yann LeCun
Well, IQ can measure approximately something for humans, but because humans come in relatively uniform form, right?

Lex Fridman
Yeah.

Yann LeCun
But it only measures one type of ability that maybe relevant for some tasks but not others. But then if you were talking about other intelligent entities for which the basic things that are easy to them is very different, then it doesn't mean anything. So intelligence is a collection of skills and an ability to acquire new skills efficiently. The collection of skills that a particular intelligent entity possess or is capable of learning quickly is different from the collection of skills of another one. Because it's a multidimensional thing, the set of skills is a high dimensional space, you can't measure, you cannot compare two things as to whether one is more intelligent than the other. It's multidimensional.

Lex Fridman
So you push back against what are called AI doomers a lot. Can you explain their perspective and why you think they're wrong?

Yann LeCun
Okay, so AI doomers imagine all kinds of catastrophe scenarios of how AI could escape or control and basically kill us all, and that relies on a whole bunch of assumptions that are mostly false. So the first assumption is that the emergence of super intelligence is going to be an event, that at some point we're going to figure out the secret and we'll turn on a machine that is super intelligent, and because we'd never done it before, it's going to take over the world and kill us all. That is false. It's not going to be an event. We're going to have systems that are as smart as a cat, have all the characteristics of human-level intelligence, but their level of intelligence would be like a cat or a parrot maybe or something. Then we're going to work our way up to make those things more intelligent. As we make them more intelligent, we're also going to put some guardrails in them and learn how to put some guardrails so they behave properly. It's not going to be one effort, that it's going to be lots of different people doing this, and some of them are going to succeed at making intelligent

systems that are controllable and safe and have the right guardrails. If some other goes rogue, then we can use the good ones to go against the rogue ones. So it's going to be my smart AI police against your rogue AI. So it's not going to be like we're going to be exposed to a single rogue AI that's going to kill us all. That's just not happening. Now, there is another fallacy, which is the fact that because the system is intelligent, it necessarily wants to take over. There is several arguments that make people scared of this, which I think are completely false as well. So one of them is in nature, it seems to be that the more intelligent species otherwise end up dominating the other and even distinguishing the others sometimes by design, sometimes just by mistake. So there is thinking by which you say, "Well, if AI systems are more intelligent than us, surely they're going to eliminate us, if not by design, simply because they don't care about us," and that's just preposterous for a number of reasons. First reason is they're not going to be a species. They're not going to be a species that competes with us. They're not going to have the desire to dominate because the desire to dominate is something that has to be hardwired into an intelligent system. It is hardwired in humans. It is hardwired in baboons, in chimpanzees, in wolves, not in orangutans. The species in which this desire to dominate or submit or attain status in other ways is specific to social species. Non-social species like orangutans don't have it, and they are as smart as we are, almost, right?

Lex Fridman
To you, there's not significant incentive for humans to encode that into the AI systems, and to the degree they do, there'll be other AIs that punish them for it, I'll compete them over it.

Yann LeCun
Well, there's all kinds of incentive to make AI systems submissive to humans.

Lex Fridman
Right.

Yann LeCun
Right? This is the way we're going to build them. So then people say, "Oh, but look at LLMs. LLMs are not controllable," and they're right. LLMs are not controllable. But objectively-driven AI, so systems that derive their answers by optimization of an objective means they have to optimize this objective, and that objective can include guardrails. One guardrail is, obey humans. Another guardrail is, don't obey humans if it's hurting other humans within limits.

Lex Fridman
Right. I've heard that before somewhere, I don't remember-

Yann LeCun
Yes, maybe in a book.

Lex Fridman
Yeah, but speaking of that book, could there be unintended consequences also from all of this?

Yann LeCun
No, of course. So this is not a simple problem. Designing those guardrails so that the system behaves properly is not going to be a simple issue for which there is a silver bullet for which you have a mathematical proof that the system can be safe. It's going to be a very progressive, iterative design system where we put those guardrails in such a way that the system behave properly. Sometimes they're going to do something that was unexpected because the guardrail wasn't right and we're dd correct them so that they do it right. The idea somehow that we can't get it slightly wrong because if we get it slightly wrong, we'll die is ridiculous. We are just going to go progressively. It is just going to be, the analogy I've used many times is turbojet design. How did we figure out how to make turbojet so unbelievably reliable? Those are incredibly complex pieces of hardware that run at really high temperatures for 20 hours at a time sometimes, and we can fly halfway around the world on a two-engine jetliner at near the speed of sound. Like how incredible is this? It's just unbelievable. Did we do this because we invented a general principle of how to make turbojets safe? No, it took decades to fine tune the design of those systems so that they were safe. Is there a separate group within General Electric or Snecma or whatever that is specialized in turbojet safety? No. The design is all about safety, because a better turbojet is also a safer turbojet, so a more reliable one. It's the same for AI. Do you need specific provisions to make AI safe? No, you need to make better AI systems, and they will be safe because they are designed to be more useful and more controllable.

Lex Fridman
So let's imagine a system, AI system that's able to be incredibly convincing and can convince you of anything. I can at least imagine such a system, and I can see such a system be weapon like because it can control people's minds. We're pretty gullible. We want to believe a thing, and you can have an AI system that controls it and you could see governments using that as a weapon. So do you think if you imagine such a system, there's any parallel to something like nuclear weapons?

Yann LeCun
No.

Lex Fridman
Why is that technology different? So you're saying there's going to be gradual development?

Yann LeCun
Yeah.

Lex Fridman
It might be-Gradual development is going to be, it might be rapid, but there'll be iterative and then we'll be able to respond and so on.

Yann LeCun
So that AI system designed by Vladimir Putin or whatever, or his minions is going to be talking to, trying to talk to every American to convince them to vote for-

Lex Fridman
Whoever.

Yann LeCun
- Whoever pleases Putin.

Lex Fridman
Sure.

Yann LeCun
Or whatever, or rile people up against each other as they've been trying to do. They're not going to be talking to you, they're going to be talking to your AI assistant, which is going to be as smart as theirs. Because as I said, in the future, every single one of your interaction with the digital world will be mediated by your AI assistant. So the first thing you're going to ask, is this a scam? Is this thing telling me the truth? It's not even going to be able to get to you because it's only going to talk to your AI system or your AI system. It's going to be like a spam filter. You're not even seeing the email, the spam email. It's automatically put in a folder that you never see. It's going to be the same thing. That AI system that tries to convince you of something is going to be talking to AI assistant, which is going to be at least as smart as it, and it's going to say, "This is spam." It's not even going to bring it to your attention.

Lex Fridman
So to you, it's very difficult for any one AI system to take such a big leap ahead to where it can convince even the other AI systems. There's always going to be this kind of race where nobody's way ahead.

Yann LeCun
That's the history of the world. History of the world is whenever there is a progress someplace, there is a countermeasure and it's a cat and mouse game.

Lex Fridman
Mostly yes, but this is why nuclear weapons are so interesting because that was such a powerful weapon that it mattered who got it first. That you could imagine Hitler, Stalin, Mao

getting the weapon first, and that having a different kind of impact on the world than the United States getting the weapon first. But to you, nuclear weapons, you don't imagine a breakthrough discovery and then Manhattan Project-like effort for AI?

Yann LeCun
No. No, as I said, it's not going to be an event. It's going to be continuous progress. And whenever one breakthrough occurs, it's going to be widely disseminated really quickly.

Lex Fridman
Yeah.

Yann LeCun
Probably first within industry. This is not a domain where government or military organizations are particularly innovative and they're in fact way behind. And so this is going to come from industry and this kind of information disseminates extremely quickly. We've seen this over the last few years where you have a new - even take AlphaGo - this was reproduced within three months even without particularly detailed information, right?

Lex Fridman
Yeah. This is an industry that's not good at secrecy. But people -

Yann LeCun
No. But even if there is, just the fact that you know that something is possible makes you realize that it's worth investing the time to actually do it. You may be the second person to do it, but you'll do it. And same for all the innovations of self supervision in transformers, decoder only architectures, LLMs. Those things, you don't need to know exactly the details of how they work to know that it's possible because it's deployed and then it's getting reproduced. And then people who work for those companies move. They go from one company to another and the information disseminates. What makes the success of the US tech industry and Silicon Valley in particular is exactly that, is because the information circulates really, really quickly and disseminates very quickly. And so the whole region is ahead because of that circulation of information.

Lex Fridman
Maybe just to linger on the psychology of AI doomers, you give, in the classic Yann LeCun way, a pretty good example of just when a new technology comes to be, you say engineer says, "I invented this new thing. I call it a ball pen." And then the Twitter sphere responds, "OMG people could write horrible things with it, like misinformation, propaganda, hate speech. Ban it now." Then writing doomers come in, akin to the AI doomers, "Imagine if everyone can get a ball pen. This could destroy society. There should be a law against using ball pen to write hate speech, regulate ball pens now." And then the pencil industry mogul says, "Yeah, ball pens are very dangerous. Unlike pencil writing, which is erasable, ball pen

writing stays forever. Government should require a license for a pen manufacturer." This does seem to be part of human psychology when it comes up against new technology. What deep insights can you speak to about this?

Yann LeCun
Well, there is a natural fear of new technology and the impact it can have in society. And people have instinctive reaction to the world they know being threatened by major transformations that are either cultural phenomena or technological revolutions. And they fear for their culture, they fear for their job, they fear for the future of their children and their way of life. So any change is feared. And you see this along history, any technological revolution or cultural phenomenon was always accompanied by groups or reaction in the media that basically attributed all the current problems of society to that particular change. Electricity was going to kill everyone at some point. The train was going to be a horrible thing because you can't breathe past 50 kilometers an hour. And so there's a wonderful website called the Pessimist Archive.

Lex Fridman
It's great.

Yann LeCun
Which has all those newspaper clips of all the horrible things people imagine would arrive because of either a technological innovation or a cultural phenomenon, just wonderful examples of jazz or comic books being blamed for unemployment or young people not wanting to work anymore and things like that. And that has existed for centuries and it's knee-jerk reactions. The question is do we embrace change or do we resist it? And what are the real dangers as opposed to the imagined ones?

Lex Fridman
So people worry about, I think one thing they worry about with big tech, something we've been talking about over and over, but I think worth mentioning again, they worry about how powerful AI will be and they worry about it being in the hands of one centralized power of just a handful of central control. And so that's the skepticism with big tech you make, these companies can make a huge amount of money and control this technology, and by so doing take advantage, abuse the little guy in society.

Yann LeCun
Well, that's exactly why we need open source platforms.

Lex Fridman
Yeah, I just wanted to nail the point home more and more.

Yann LeCun
Yes.

Lex Fridman
So let me ask you on your, like I said, you do get a little bit flavorful on the internet. Joscha Bach tweeted something that you LOL'd at in reference to HAL 9,000. Quote, "I appreciate your argument and I fully understand your frustration, but whether the pod bay doors should be opened or closed is a complex and nuanced issue." So you're at the head of Meta AI. This is something that really worries me, that our AI overlords will speak down to us with corporate speak of this nature, and you resist that with your way of being. Is this something you can just comment on, working at a big company, how you can avoid the over fearing, I suppose, through caution create harm?

Yann LeCun
Yeah. Again, I think the answer to this is open source platforms and then enabling a widely diverse set of people to build AI assistance that represent the diversity of cultures, opinions, languages, and value systems across the world so that you're not bound to just be brainwashed by a particular way of thinking because of a single AI entity. So, I think it's a really, really important question for society. And the problem I'm seeing is that, which is why I've been so vocal and sometimes a little sardonic about it-

Lex Fridman
Never stop. Never stop, Yann. We love it.

Yann LeCun
- is because I see the danger of this concentration of power through proprietary AI systems as a much bigger danger than everything else. That if we really want diversity of opinion AI systems, that in the future where we'll all be interacting through AI systems, we need those to be diverse for the preservation of diversity of ideas and creed and political opinions and whatever, and the preservation of democracy. And what works against this is people who think that for reasons of security, we should keep the AI systems under lock and key because it's too dangerous to put it in the hands of everybody, because it could be used by terrorists or something. That would lead to potentially a very bad future in which all of our information diet is controlled by a small number of companies through proprietary systems.

Lex Fridman
So you trust humans with this technology to build systems that are on the whole good for humanity.

Yann LeCun
Isn't that what democracy and free speech is all about?

Lex Fridman
I think so.

Yann LeCun
Do you trust institutions to do the right thing?

Lex Fridman
Sure.

Yann LeCun
Do you trust people to do the right thing? And yeah, there's bad people who are going to do bad things, but they're not going to have superior technology to the good people. So then it's going to be my good AI against your bad AI, right? There's the examples that we were just talking about of maybe some rogue country will build some AI system that's going to try to convince everybody to go into a civil war or something or elect a favorable ruler, but then they will have to go past our AI systems.

Lex Fridman
Right. An AI system with a strong Russian accent will be trying to convince our-

Yann LeCun
And doesn't put any articles in their sentences.

Lex Fridman
Well, it'll be at the very least, absurdly comedic. Okay. So since we talked about the physical reality, I'd love to ask your vision of the future with robots in this physical reality. So many of the kinds of intelligence that you've been speaking about would empower robots to be more effective collaborators with us humans. So since Tesla's Optimus team has been showing us some progress on humanoid robots, I think it really reinvigorated the whole industry that I think Boston Dynamics has been leading for a very, very long time. So now there's all kinds of companies Figure AI, obviously Boston Dynamics.

Yann LeCun
Unitree.

Lex Fridman
Unitree, but there's a lot of them.

Yann LeCun
There's a few of them.

Lex Fridman
It's great. It's great. I love it. So do you think there'll be millions of humanoid robots walking around soon?

Yann LeCun
Not soon, but it's going to happen. The next decade I think is going to be really interesting in robots, the emergence of the robotics industry has been in the waiting for 10, 20 years without really emerging other than for pre-program behavior and stuff like that. And the main issue is, again, the Moravec paradox, how do we get those systems to understand how the world works and plan actions? And so we can do it for really specialized tasks. And the way Boston Dynamics goes about it is basically with a lot of handcrafted dynamical models and careful planning in advance, which is very classical robotics with a lot of innovation, a little bit of perception, but it's still not, they can't build a domestic robot. We're still some distance away from completely autonomous level five driving, and we're certainly very far away from having level five autonomous driving by a system that can train itself by driving 20 hours like any 17-year-old. So until we have, again, world models, systems that can train themselves to understand how the world works, we're not going to have significant progress in robotics. So a lot of the people working on robotic hardware at the moment are betting or banking on the fact that AI is going to make sufficient progress towards that,

Lex Fridman
And they're hoping to discover a product in it too. Because before you have a really strong world model, there'll be an almost strong world model and people are trying to find a product in a clumsy robot, I suppose, not a perfectly efficient robot. So there's the factory setting where humanoid robots can help automate some aspects of the factory. I think that's a crazy difficult task because of all the safety required and all this kind of stuff. I think in the home is more interesting, but then you start to think, I think you mentioned loading the dishwasher, right?

Yann LeCun
Yeah.

Lex Fridman
I suppose that's one of the main problems you're working on.

Yann LeCun
There's cleaning up, cleaning the house, clearing up the table after a meal.

Lex Fridman
Sure.

Yann LeCun
Washing the dishes, all those tasks, cooking. All the tasks that in principle could be automated but are actually incredibly sophisticated, really complicated.

Lex Fridman
But even just basic navigation around a space full of uncertainty.

Yann LeCun
That works. You can do this now, navigation is fine.

Lex Fridman
Well, navigation in a way that's compelling to us humans is a different thing.

Yann LeCun
Yeah, it's not going to be necessarily. We have demos actually, because there is a so-called embodied AI group at fair, and they've been not building their own robots, but using commercial robots. And you can tell the robot dog go to the fridge and they can actually open the fridge and they can probably pick up a can in the fridge and stuff like that and bring it to you. So it can navigate, it can grab objects as long as it's been trained to recognize them, which vision systems work pretty well nowadays, but it's not like a completely general robot that would be sophisticated enough to do things like clearing up the dinner table.

Lex Fridman
To me, that's an exciting future of getting humanoid robots, robots in general in the home more and more, because it gets humans to really directly interact with AI systems in the physical space. And in so doing it allows us to philosophically, psychologically explore our relationships with robots. Going to be really, really, really interesting. So I hope you make progress on the whole JEPA thing soon.

Yann LeCun
Well, I hope things can work as planned. Again, we've been working on this idea of self-supervised learning from video for 10 years, and only made significant progress in the last two or three.

Lex Fridman
And actually you've mentioned that there's a lot of interesting breakage that can happen without having access to a lot of compute. So if you're interested in doing a PhD in this kind of stuff, there's a lot of possibilities still to do innovative work. So what advice would you give to an undergrad that's looking to go to grad school and do a PhD?

Yann LeCun
Basically, I've listed them already, this idea of how do you train a world model by observation? And you don't have to train necessarily on gigantic data sets. It could turn out to be necessary, to actually train on large data sets, to have emergent properties like we have with other lamps. But I think there is a lot of good ideas that can be done without necessarily scaling up than there is how do you do planning with a learn world model? If the world the system evolves in is not the physical world, but is the world of let's say the internet or some sort of world where an action consists in doing a search in a search engine, or interrogating a database, or learning a simulation, or calling a calculator, or solving a differential equation, how do you get a system to actually plan a sequence of actions to give the solution to a problem? And so the question of planning is not just a question of planning physical actions. It could be planning actions to use tools for a dialogue system or for any kind of intelligence system. And there's some work on this, but not a huge amount. Some work at fair, one called Toolformer, which was a couple years ago and some more recent work on planning, but I don't think we have a good solution for any of that. Then there is the question of hierarchical planning. So the example I mentioned of planning a trip from New York to Paris, that's hierarchical, but almost every action that we take involves hierarchical planning in some sense, and we really have absolutely no idea how to do this. There's zero demonstration of hierarchical planning in AI where the various levels of representations that are necessary have been learned. We can do two level hierarchical planning when we designed the two levels. So for example, you have a dog-like robot, you want it to go from the living room to the kitchen. You can plan a path that avoids the obstacle, and then you can send this to a lower level planner that figures out how to move the legs to follow that trajectories. So that works, but that two level planning is designed by hand. We specify what the proper levels of abstraction, the representation at each level of abstraction have to be. How do you learn this? How do you learn that hierarchical representation of action plans? With convolutional deep learning, we can train the system to learn hierarchical representations of percepts. What is the equivalent when what you're trying to represent are action plans?

Lex Fridman
For action plans, yeah. So you want basically a robot dog or humanoid robot that turns on and travels from New York to Paris all by itself.

Yann LeCun
For example.

Lex Fridman
It might have some trouble at the TSA.

Yann LeCun
No, but even doing something fairly simple like a household task, like cooking or something.

Lex Fridman
Yeah, there's a lot involved. It's a super complex task and once again, we take it for granted. What hope do you have for the future of humanity? We're talking about so many exciting technologies, so many exciting possibilities. What gives you hope when you look out over the next 10, 20, 50, a hundred years? If you look at social media, there's wars going on, there's division, there's hatred, all this kind of stuff that's also part of humanity. But amidst all that, what gives you hope?

Yann LeCun
I love that question. We can make humanity smarter with AI. AI basically will amplify human intelligence. It's as if every one of us will have a staff of smart AI assistants. They might be smarter than us. They'll do our bidding, perhaps execute a task in ways that are much better than we could do ourselves, because they'd be smarter than us. And so it's like everyone would be the boss of a staff of super smart virtual people. So we shouldn't feel threatened by this any more than we should feel threatened by being the manager of a group of people, some of whom are more intelligent than us. I certainly have a lot of experience with this, of having people working with me who are smarter than me. That's actually a wonderful thing. So having machines that are smarter than us, that assist us in all of our tasks, our daily lives, whether it's professional or personal, I think would be an absolutely wonderful thing. Because intelligence is the commodity that is most in demand. That's really what I mean. All the mistakes that humanity makes is because of lack of intelligence really, or lack of knowledge, which is related. So making people smarter, we just can only be better. For the same reason that public education is a good thing and books are a good thing, and the internet is also a good thing, intrinsically and even social networks are a good thing if you run them properly. It's difficult, but you can. Because it helps the communication of information and knowledge and the transmission of knowledge. So AI is going to make humanity smarter. And the analogy I've been using is the fact that perhaps an equivalent event in the history of humanity to what might be provided by generalization of AI assistant is the invention of the printing press. It made everybody smarter, the fact that people could have access to books. Books were a lot cheaper than they were before, and so a lot more people had an incentive to learn to read, which wasn't the case before. And people became smarter. It enabled the enlightenment. There wouldn't be an enlightenment without the printing press. It enabled philosophy, rationalism, escape from religious doctrine, democracy, science. And certainly without this, there wouldn't have been the American Revolution or the French Revolution. And so we would still be under a feudal regimes perhaps. And so it completely transformed the world because people became smarter and learned about things. Now, it also created 200 years of essentially religious conflicts in Europe because the first thing that people read was the Bible and realized that perhaps there was a different interpretation of the Bible than what the priests were telling them. And so that created the Protestant movement and created the rift. And in fact, the Catholic Church didn't like the idea of the printing press, but they had no choice. And so it had some bad effects and some good effects. I don't think anyone today would say that the invention

of the printing press had a overall negative effect despite the fact that it created 200 years of religious conflicts in Europe. Now, compare this, and I thought I was very proud of myself to come up with this analogy, but realized someone else came with the same idea before me, compare this with what happened in the Ottoman Empire. The Ottoman Empire banned the printing press for 200 years, and he didn't ban it for all languages, only for Arabic. You could actually print books in Latin or Hebrew or whatever in the Ottoman Empire, just not in Arabic. And I thought it was because the rulers just wanted to preserve the control over the population and the religious dogma and everything. But after talking with the UAE Minister of AI, Omar Al Olama, he told me no, there was another reason. And the other reason was that it was to preserve the corporation of calligraphers. There's an art form, which is writing those beautiful Arabic poems or whatever, religious text in this thing. And it was a very powerful corporation of scribes basically that run a big chunk of the empire, and we couldn't put them out of business. So they banned the printing press in part to protect that business. Now, what's the analogy for AI today? Who are we protecting by banning AI? Who are the people who are asking that AI be regulated to protect their jobs? And of course, it's a real question of what is going to be the effect of a technological transformation like AI on the job market and the labor market? And there are economists who are much more expert at this than I am, but when I talk to them, they tell us we're not going to run out of the job. This is not going to cause mass unemployment. This is just going to be gradual shift of different professions. The professions that are going to be hot 10 or 15 years from now, we have no idea today what they're going to be. The same way, if you go back 20 years in the past, who could have thought 20 years ago that the hottest job - even 5, 10 years ago - was mobile app developer? Smartphones weren't invented.

Lex Fridman
Most of the jobs of the future might be in the metaverse.

Yann LeCun
Well, it could be, yeah.

Lex Fridman
But the point is you can't possibly predict. But you're right. You made a lot of strong points. And I believe that people are fundamentally good. And so if AI, especially open source AI, can make them smarter, it just empowers the goodness in humans.

Yann LeCun
So I share that feeling, I think people are fundamentally good. And in fact, a lot of doomers are doomers because they don't think that people are fundamentally good, and they either don't trust people or they don't trust the institution to do the right thing so that people behave properly.

Lex Fridman
Well, I think both you and I believe in humanity, and I think I speak for a lot of people in saying thank you for pushing the open source movement, pushing to making both research and AI open source, making it available to people, and also the models themselves, making it open source. So thank you for that. And thank you for speaking your mind in such colorful and beautiful ways on the internet. I hope you never stop. You're one of the most fun people I know and get to be a fan of. So Yann, thank you for speaking to me once again, and thank you for being you.

Yann LeCun
Thank you, Lex.