The following is a conversation with George Hotz,
his third time on this podcast.
He's the founder of Comma AI
that seeks to solve autonomous driving
and is the founder of a new company called TinyCorp
that created TinyGrad, a neural network framework
that is extremely simple with the goal
of making it run on any device by any human easily
and efficiently.
As you know, George also did a large number
of fun and amazing things from hacking the iPhone
to recently joining Twitter for a bit
as an intern in quotes,
making the case for refactoring the Twitter code base.
In general, he's a fascinating engineer and human being
and one of my favorite people to talk to.
And now a quick few second mention of his sponsor.
Check them out in the description.
It's the best way to support this podcast.
We've got Numeri for the world's hardest
data science tournament, Babbel for learning new languages,
Netsuite for business management software,
Insight Tracker for blood paneling
and AG1 for my daily multivitamin.
Choose wisely, my friends.
Also, if you want to work on our team,
we're always hiring, go to lexfreedman.com slash hiring.
And now onto the full ad reads.
As always, no ads in the middle.
I try to make this interesting,
but if you must skip them, friends,
please still check out our sponsors.
I enjoy their stuff.
Maybe you will too.
This episode is brought to you by Numeri,
a hedge fund that uses artificial intelligence
and machine learning to make investment decisions.
They created a tournament that challenges data scientists
to build best predictive models for financial markets.
It's basically just a really, really difficult,
real world data set to test out your ideas
for how to build machine learning models.
I think this is a great educational platform.

I think this is a great way to explore,
to learn about machine learning,
to really test yourself on real world data
with the consequences.
No financial background is needed.
The models are scored based on
how well they perform on unseen data.
And the top performers receive a share
of the tournament's prize pool.
Head over to Numeri slash lex,
that's N-U-M-E-R dot A-I slash lex
to sign up for a tournament
and hone your machine learning skills.
That's Numeri slash lex for a chance to play against me
and win the share of the tournament's prize pool.
That's Numeri slash lex.
This show is also brought to you by Babbel,
an app and website that gets you speaking
in a new language within weeks.
I have been using it to learn a few languages,
Spanish, to review Russian, to practice Russian,
to revisit Russian from a different perspective
because that becomes more and more relevant
for some of the previous conversations I've had
and some upcoming conversations I have.
It really is fascinating how much another language,
knowing another language,
even to a degree where you can just have little bits
and pieces of a conversation can really unlock an experience
in another part of the world.
When you travel in France and Paris,
just having a few words at your disposal, a few phrases,
it begins to really open you up
to strange, fascinating new experiences
that ultimately, at least to me,
teach me that we're all the same.
We have to first see our differences
to realize those differences are grounded
in a basic humanity.
And that experience, that we're all very different
and yet at the core the same.
I think travel with aid of language really helps unlock.
You can get 55% off your Babbel subscription

at babbel.com slash lexpod
that's spelled B-A-B-B-E-L.com slash lexpod.
Rules and restrictions apply.
This show is also brought to you by Netsuite,
an all-in-one cloud business management system.
They manage all the messy stuff
that is required to run a business, the financials,
the human resources, the inventory,
if you do that kind of thing, e-commerce, all that stuff,
all the business-related details.
I know how stressed I am about everything that's required
to run a team, to run a business
that involves much more than just ideas
and designs and engineering
and involves all the management of human beings,
all the complexities of that, the financials, all of it
and sure you should be using the best tools for the job.
I sometimes wonder if I have it in me mentally
and skill-wise to be a part of running a large company.
I think like with a lot of things in life,
it's one of those things you shouldn't wonder too much about.
You should either do or not do.
But again, using the best tools for the job
is required here.
You can start now with a no payment or interest
for six months, go to netsuite.com slash lex
to access their one-of-a-kind financing program.
That's netsuite.com slash lex.
This show is also brought to you by Insight Tracker,
a service I use to track biological data,
data that comes from my body to predict,
to tell me what I should do with my lifestyle,
with my diet, what's working and what's not working.
It's obvious, all the exciting breakthroughs
that are happening with Transformers,
with large language models, even with diffusion,
all of that is obvious that with raw data,
with huge amounts of raw data,
fine-tuned to the individual,
would really reveal to us the signal
in all the noise of biology.
I feel like that's on the horizon.
The kinds of leaps in development that we saw in language

and now more and more visual data.
I feel like biological data is around the corner,
unlocking what's there in this multi-hierarchical,
distributed system that is our biology.
What is it telling us?
What is the secrets it holds?
What is the thing that it's missing that could be aided?
Simple lifestyle changes, simple diet changes,
simple changes in all kinds of things
that are controllable by individual human being.
I can't wait till that's a possibility
and Insight Tracker is taking steps towards that.
Get special savings for a limited time
when you go to insitracker.com slash lex.
This show is also brought to you by Athletic Greens
that's now called AG1.
It has the AG1 drink.
I drink it twice a day.
At the very least, it's an all-in-one daily drink
to support better health and peak performance.
I drink it cold, it's refreshing, it's grounding.
It helps me reconnect with the basics,
the nutritional basics that makes this whole machine
that is our human body run.
All the crazy mental stuff I do for work,
the physical challenges, everything,
the highs and lows of life itself.
All of that is somehow made better knowing that
at least you got your nutrition in check.
At least you're getting enough sleep.
At least you're doing the basics.
At least you're doing the exercise.
Once you get those basics in place,
I think you can do some quite difficult things in life.
But anyway, beyond all that is just a source of happiness
and a kind of a feeling of home.
The feeling that comes from returning to the habit
time and time again.
Anyway, they'll give you one month supply of fish oil
when you sign up at drinkag1.com slash Lex.
This is the Lex Friedman Podcast.
To support it, please check out our sponsors
in the description.

And now, dear friends, here's George Hottz.
["The Time Is An Illusion"]
You mentioned something in a stream
about the philosophical nature of time.
So let's start with a wild question.
Do you think time is an illusion?
You know, I sell phone calls to Kamma for $1,000.
And some guy called me and like, you know, it's $1,000.
You can talk to me for half an hour.
And he's like, yeah, okay.
So like, time doesn't exist.
And I really wanted to share this with you.
I'm like, oh, what do you mean time doesn't exist, right?
Like, I think time is a useful model.
Whether it exists or not, right?
Like, does quantum physics exist?
Well, it doesn't matter.
It's about whether it's a useful model to describe reality.
Is time maybe compressive?
Do you think there is an objective reality
or is everything just useful models?
Like underneath it all, is there an actual thing
that we're constructing models for?
I don't know.
I was hoping you would know.
I don't think it matters.
I mean, this kind of connects to the models
of constructive reality with machine learning, right?
Sure.
Like, is it just nice to have useful approximations
of the world such that we can do something with it?
So there are things that are real.
Colomagraph complexity is real.
Yeah.
Yeah, the compressive thing, math is real.
Yeah.
This should be a T-shirt.
And I think hard things are actually hard.
I don't think P equals NP.
Ooh, strong words.
Well, I think that's the majority.
I do think factoring is in P, but.
I don't think you're the person

that falls the majority in all walks of life.
So, but it's good.
Well, for that one I do.
Yeah.
In theoretical computer science, you're one of the sheep.
All right.
But do you, time is a useful model?
Sure.
Hmm.
What were you talking about on the stream with time?
Are you made of time?
If I remembered half the things I said on stream,
someday someone's gonna make a model of all of that
and it's gonna come back to haunt me.
Someday soon?
Yeah, probably.
Would that be exciting to you or sad
that there's a George Hott's model?
I mean, the question is when the George Hott's model
is better than George Hott's.
Like, I am declining and the model is growing.
What is the metric by which you measure better or worse
in that if you're competing with yourself?
Maybe you can just play a game
where you have the George Hott's answer
and the George Hott's model answer
and ask which people prefer.
People close to you or strangers?
Either one, it will hurt more
when it's people close to me,
but both will be overtaken by the George Hott's model.
It'd be quite painful, right?
Loved ones, family members,
would rather have the model
over for Thanksgiving than you.
Yeah.
Or like significant others,
would rather sext
with the large language model version of you.
Especially when it's fine tuned to their preferences.
Is it?
Yeah.
Well, that's what we're doing in a relationship, right?

We're just fine tuning ourselves,
but we're inefficient with it
because we're selfish and greedy and so on.
Language models can fine tune more efficiently,
more selflessly.
There's a Star Trek Voyager episode
where Catherine Janeway lost in the Delta Quadrant
makes herself a lover on the holodeck.
And the lover falls asleep on her arm
and he snores a little bit
and Janeway edits the program to remove that.
And then of course the realization is,
wait, this person's terrible.
It is actually all their nuances and quirks
and slight annoyances that make this relationship worthwhile.
But I don't think we're gonna realize that
until it's too late.
Well, I think a large language model
could incorporate the flaws and the quirks
and all that kind of stuff.
Just the perfect amount of quirks and flaws
to make you charming without crossing the line.
Yeah.
Yeah.
And there's probably a good like approximation
of the like the percent of time
the language model should be cranky
or an asshole or jealous or all this kind of stuff.
And of course it can and it will
but all that difficulty at that point is artificial.
There's no more real difficulty.
Okay, what's the difference in real and artificial?
Artificial difficulty is difficulty
that's like constructed or could be turned off with a knob.
Real difficulty is like you're in the woods
and you gotta survive.
So if something can not be turned off with a knob,
it's real?
Yeah, I think so.
Or I mean, you can't get out of this
by smashing the knob with a hammer.
I mean, maybe you kind of can, you know,
into the wild when, you know, Alexander Supertramp,

he wants to explore something
that's never been explored before,
but it's the 90s, everything's been explored.
So he's like, well, I'm just not gonna bring a map.
I mean, no, you're not exploring.
You should have brought a map, dude.
You died.
There was a bridge a mile from where you were camping.
How does that connect to the metaphor of the knob?
By not bringing the map, you didn't become an explorer.
You just smashed the thing.
Yeah.
Yeah, the difficulty is still artificial.
You failed before you started.
What if we just don't have access to the knob?
Well, that maybe is even scarier, right?
Like we already exist in a world of nature
has been fine-tuned over billions of years
to have humans build something
and then throw the knob away
and some grand romantic gesture is horrifying.
Do you think of us humans as individuals
that are like born and die,
or is it, are we just all part of one living organism
that is Earth, that is nature?
I don't think there's a clear line there.
I think it's all kind of just fuzzy.
I don't know.
I mean, I don't think I'm conscious.
I don't think I'm anything.
I think I'm just a computer program.
So it's all computation.
But I think running your head is just a computation.
Everything running in the universe is computation,
I think, I believe the extended church trying thesis.
Yeah, but there seems to be an embodiment
to your particular computation.
Like there's a consistency.
Well, yeah, but I mean, models have consistency too.
Yeah.
Models that have been RLHF will continually say,
like, well, how do I murder ethnic minorities?
Oh, well, I can't let you do that, Al.

There's a consistency to that behavior.
So RLHF, like we are RLHF each other.
We find, we provide human feedback
and thereby fine-tune these little pockets of computation,
but it's still unclear why that pocket of computation
stays with you, like for years.
It just kind of fall, like you have this consistent set
of physics, biology, what, like whatever you call
the neurons firing, like the electrical signals
and the mechanical signals, all of that,
that seems to stay there and it contains information,
it stores information and that information
permeates through time and stays with you.
There's like memory, there's like sticky.
Okay, to be fair, like a lot of the models
we're building today are very, even RLHF is
nowhere near as complex as the human loss function.
Reinforcement learning with human feedback.
You know, when I talked about will GPT-12 be AGI,
my answer is no, of course not.
I mean, cross-entropy loss is never gonna get you there.
You need probably RL in fancy environments
in order to get something that would be considered like AGI-like.
So to ask like the question about like why, I don't know,
like it's just some quirk of evolution, right?
I don't think there's anything particularly special
about where I ended up, where humans ended up.
It's okay, we have human level intelligence.
Would you call that AGI?
Whatever we have is GI?
Look, actually I don't really even like the word AGI,
but general intelligence is defined to be
whatever humans have.
Okay, so why can GPT-12 not get us to AGI?
Can we just like linger on that?
If your loss function is categorical cross-entropy,
if your loss function is just try to maximize compression.
I have a SoundCloud, I rap and I tried to get chat GPT
to help me write raps.
And the raps that it wrote sounded like YouTube comment raps.
You know, you can go on any rap beat online
and you can see what people put in the comments.
And it's the most like mid-quality rap you can find.

Is mid good or bad?
Mid is bad.
Mid is bad.
It's like mid, it's like...
Every time I talk to you, I learn new words.
Mid, mid.
Yeah.
I was like, is it like basic?
Is that what mid means?
Kind of, it's like middle of the curve, right?
So there's like, I like see that intelligence curve.
And you have like the dumb guy, the smart guy,
and then the mid guy, actually being the mid guy is the worst.
The smart guy is like, I put all my money in Bitcoin.
The mid guy is like, you can't put money in Bitcoin,
that's not real money.
And all of it is a genius meme.
That's another interesting one.
Memes, the humor, the idea, the absurdity
encapsulated in a single image.
And it just kind of propagates virally
between all of our brains.
I didn't get much sleep last night,
so I'm very, I sound like I'm high, but I swear I'm not.
Do you think we have ideas or ideas have us?
I think that we're gonna get super scary memes
once the AIs actually are superhuman.
Ooh, you think AI will generate memes?
Of course.
You think it'll make humans laugh?
I think it's worse than that.
So, Infinite Jest, it's introduced in the first 50 pages,
is about a tape that you, once you watch it once,
you only ever wanna watch that tape.
In fact, you wanna watch the tape so much
that someone says, okay, here's a hacksaw,
cut off your pinky, and then I'll let you watch the tape
again, and you'll do it.
So, we're actually gonna build that, I think.
But it's not gonna be one static tape.
I think the human brain is too complex
to be stuck in one static tape like that.
If you look at like ant brains,

maybe they can be stuck on a static tape.
But we're going to build that using generative models.
We're going to build the TikTok
that you actually can't look away from.
So, TikTok is already pretty close there,
but the generation is done by humans.
The algorithm is just doing their recommendation.
But if the algorithm is also able to do the generation.
Well, it's a question about how much
intelligence is behind it, right?
So, the content is being generated
by, let's say, one humanity worth of intelligence.
And you can quantify a humanity, right?
That's a, you know, it's exoflops, yadaflops.
But you can quantify it.
Once that generation is being done by 100 humanities,
you're done.
This is actually scale, that's the problem.
But also speed.
Yeah.
And what if it's sort of manipulating
the very limited human dopamine engine for porn?
Imagine just TikTok, but for porn.
Yeah.
That's like a brave new world.
I don't even know what it'll look like, right?
Like, again, you can't imagine the behaviors
of something smarter than you.
But a super intelligent and an agent
that just dominates your intelligence so much
will be able to completely manipulate you.
Is it possible that it won't really manipulate?
It'll just move past us.
It'll just kind of exist the way water exists
or the air exists.
You see, and that's the whole AI safety thing.
It's not the machine that's gonna do that.
It's other humans using the machine
that are gonna do that to you.
Yeah.
Because the machine is not interested in hurting humans.
The machine is a machine.
But the human gets the machine

and there's a lot of humans out there
very interested in manipulating you.
Well, let me bring up Eliezer Yatkowski,
who recently sat where you're sitting.
He thinks that AI will almost surely kill everyone.
Do you agree with him or not?
Yes, but maybe for a different reason.
Okay.
And then I'll try to get you to find hope
or we could find a note to that answer, but why yes?
Okay, why didn't nuclear weapons kill everyone?
That's a good question.
I think there's an answer.
I think it's actually very hard
to deploy nuclear weapons tactically.
It's very hard to accomplish tactical objectives.
Great, I can nuke their country.
When I have an irradiated pile of rubble, I don't want that.
Why not?
Why don't I want an irradiated pile of rubble?
Well, the reason is no one wants
an irradiated pile of rubble.
Oh, because you can't use that land for resources.
You can't populate the land.
Yeah, what you want a total victory in a war
is not usually the irradiation
and eradication of the people there.
It's the subjugation and domination of the people.
Okay, so you can't use this strategically,
tactically in a war to help gain a military advantage.
It's all complete destruction.
All right, but there's egos involved.
It's still surprising.
Still surprising that nobody pressed a big red button.
It's somewhat surprising,
but you see it's the little red button
that's gonna be pressed with AI that's gonna,
and that's why we die.
It's not because the AI,
if there's anything in the nature of AI,
it's just an nature of humanity.
What's the algorithm behind the little red button?
What possible ideas do you have

for how human species ends?
Sure, so I think the most obvious way to me is wireheading.
We end up amusing ourselves to death.
We end up all staring at that infinite TikTok
and forgetting to eat.
Maybe it's even more benign than this.
Maybe we all just stop reproducing.
No, to be fair,
it's probably hard to get all of humanity.
Yeah.
It probably-
The interesting thing about humanity is the diversity in it.
Oh yeah.
Organisms in general.
There's a lot of weirdos out there.
Well-
Two of them are sitting here.
I mean, diversity in humanity is-
We do respect.
I wish I was more weird.
No, I'm kind of, look, I'm drinking smart water, man.
That's like a Coca-Cola product, right?
Do you want corporate, George Haas?
Yeah, I want corporate.
No, the amount of diversity in humanity,
I think is decreasing,
just like all the other biodiversity on the planet.
Oh boy, yeah.
Right?
Social media is not helping, huh?
Go eat McDonald's in China.
Yeah.
Yeah, no, it's the interconnectedness
that's doing it.
Oh, that's interesting.
So everybody starts relying on the connectivity
of the internet and over time that reduces the diversity,
the intellectual diversity,
and then that gets you everybody into a funnel.
There's still going to be a guy in Texas.
There is, and yeah.
A bunker.
To be fair, do I think AI kills us all?

I think AI kills everything we call society today.
I do not think it actually kills the human species.
I think that's actually incredibly hard to do.
Yeah, but society,
like if we start over, that's tricky.
Most of us don't know how to do most things.
Yeah, but some of us do.
And they'll be okay and they'll rebuild
after the great AI.
What's rebuilding look like?
How far, like, how much do we lose?
Like, what has human civilization done?
That's interesting.
A combustion engine, electricity.
So power and energy, that's interesting.
Like how to harness energy.
Whoa, whoa, whoa, they're going to be
religiously against that.
Are they going to get back to, like, fire?
Sure.
I mean, they'll be, it'll be like, you know,
some kind of Amish-looking kind of thing, I think.
I think they're going to have very strong
taboos against technology.
Like, technology is almost like a new religion.
Technology is the devil.
And nature is God.
Sure.
It's closer to nature.
But can you really get away from AI?
If it destroyed 99% of the human species,
isn't it somehow have a hold, like a strong hold?
Well, what's interesting about everything we build,
I think we're going to build super intelligence
before we build any sort of robustness in the AI.
We cannot build an AI that is capable
of going out into nature and surviving
like a bird, right?
A bird is an incredibly robust organism.
We've built nothing like this.
We haven't built a machine that's capable of reproducing.
Yes.
But there is, you know, I work with Lego robots a lot now.

I have a bunch of them.
They're mobile.
They can't reproduce, but all they need is,
I guess you're saying they can't repair themselves.
But if you have a large number,
if you have like a hundred million of them.
Let's just focus on them reproducing, right?
Do they have microchips in them?
Okay.
Then do they include a fab?
No.
Then how are they going to reproduce?
Well, they're, it doesn't have to be all on board, right?
They can go to a factory, to a repair shop.
Yeah, but then you're really moving away from robustness.
Yes.
All of life is capable of reproducing
without needing to go to a repair shop.
Life will continue to reproduce
in the complete absence of civilization.
Robots will not.
So when the, if the AI apocalypse happens,
I mean, the AI's are going to probably die out
because I think we're going to get, again,
super intelligence long before we get robustness.
What about if you just improve the fab
to where you just have a 3D printer
that can always help you?
Well, that'd be very interesting.
I'm interested in building that.
Of course you are.
You think, how difficult is that problem
to have a robot that basically can build itself?
Very, very hard.
I think you've mentioned this like to me or somewhere
where people think it's easy conceptually.
And then they remember that you're going to have to have a fab.
Yeah, on board.
Of course.
So 3D printer that prints a 3D printer.
Yeah.
Yeah, on legs.
Why is that hard?

Well, because it's not, I mean, a 3D printer
is a very simple machine, right?
Okay, you're going to print chips.
You're going to have an atomic printer.
How are you going to dope the silicon?
Yeah.
All right.
How are you going to etch the silicon?
You're going to have to have a very interesting kind of fab
if you want to have a lot of computation on board.
They can do like structural type of robots that are dumb.
Yeah, but structural type of robots
aren't going to have the intelligence required
to survive in any complex environment.
What about like ants type of systems?
We have like trillions of them.
I don't think this works.
I mean, again, like ants at their very core
are made up of cells that are capable
of individually reproducing.
They're doing quite a lot of computation
that we're taking for granted.
It's not even just the computation.
It's that reproduction is so inherent.
Okay, so like there's two stacks of life in the world.
There's the biological stack and the silicon stack.
The biological stack starts with reproduction.
Reproduction is at the absolute core.
The first proto RNA organisms were capable of reproducing.
The silicon stack, just by as far as it's come,
is nowhere near being able to reproduce.
Yeah, so the fab movement, digital fabrication,
fabrication in the full range of what that means
is still in the early stages.
Yeah.
You're interested in this world.
Even if you did put a fab on the machine, right?
Let's say, okay, we can build fabs.
We know how to do that as humanity.
We can probably put all the precursors
that build all the machines in the fabs also in the machine.
So first off, this machine is gonna be absolutely massive.
I mean, we almost have a, like think of the size

of the thing required to reproduce a machine today, right?
Like, is our civilization capable of reproduction?
Can we reproduce our civilization on Mars?
If we were to construct a machine
that is made up of humans, like a company,
it can reproduce itself.
Yeah.
I don't know.
It feels like, like 115 people.
It gets so much harder than that.
120?
I just wanna keep our number.
I believe the Twitter can be run by 50 people.
I think that this is gonna take most of,
like it's just most of society, right?
Like we live in one globalized world.
No, but you're not interested in running Twitter.
You're interested in seeding.
Like you want to seed a civilization then,
cause humans can like have sex.
Oh, okay.
You're talking about, yeah, okay.
So you're talking about the humans reproducing
and like basically like what's the smallest
self-sustaining colony of humans?
Yeah.
Yeah, okay, fine.
But they're not gonna be making five nanometer chips.
Over time they will.
I think you're being, like we have to expand
our conception of time here.
Going back to the original time scale.
I mean, over across maybe a hundred generations,
we're back to making chips.
No, if you seed the colony correctly.
Maybe, or maybe they'll watch our colony die out over here
and be like, we're not making chips.
Don't make chips.
No, but you have to seed that colony correctly.
Whatever you do, don't make chips.
Chips are what led to their downfall.
Well, that is the thing that humans do.
They come up, they construct a devil,

a good thing and a bad thing.
And they really stick by that.
And then they murder each other over that.
There's always one asshole in the room
who murders everybody.
And usually makes tattoos and nice branding.
You need that asshole, that's the question, right?
Humanity works really hard today to get rid of that asshole,
but I think they might be important.
Yeah, this whole freedom of speech thing.
It's the freedom of being an asshole
seems kind of important.
That's right.
Man, this thing, this fab, this human fab
that we constructed, this human civilization
is pretty interesting.
And now it's building artificial copies of itself
or artificial copies of various aspects of itself
that seem interesting, like intelligence.
And I wonder where that goes.
I like to think it's just like another stack for life.
Like we have like the bio stack life,
like we're a bio stack life
and then the silicon stack life.
But it seems like the ceiling,
or there might not be a ceiling
or at least the ceiling is much higher
for the silicon stack.
Oh no, we don't know what the ceiling is
for the bio stack either, the bio stack.
The bio stack just seem to move slower.
You have Moore's law, which is not dead
despite many proclamations.
In the bio stack or the silicon?
In the silicon stack.
And you don't have anything like this in the bio stack.
So I have a meme that I posted.
I tried to make a meme, it didn't work too well.
But I posted a picture of Ronald Reagan and Joe Biden
and you look, this is 1980 and this is 2020.
And these two humans are basically like the same, right?
There's no, like there, there's been no change
in humans in the last 40 years.

And then I posted a computer from 1980
and a computer from 2020.
Wow.
Yeah, with their early stages, right?
Which is why you said when you said the fab,
the size of the fab required to make another fab
is like very large right now.
Oh yeah.
But computers were very large 80 years ago.
And they got pretty tiny.
And people are starting to want to wear them on their face
in order to escape reality.
That's the thing, in order to live inside the computer.
Yeah.
Put a screen right here.
I don't have to see the rest of the US halls.
I've been ready for a long time.
You like virtual reality?
I love it.
Do you want to live there?
Yeah.
Yeah, part of me does too.
How far away are we, do you think?
Judging from what you can buy today far, very far.
I got to tell you that I had the experience
of Meta's Kodak Avatar,
where it's an ultra high resolution scan.
It looked real.
I mean, the headsets just are not quite
at like eye resolution yet.
I haven't put on any headset where I'm like,
oh, this could be the real world.
Whereas when I put good headphones on, audio is there.
I like, we can reproduce audio
that I'm like, I'm actually in a jungle right now.
If I close my eyes, I can't tell I'm not.
Yeah.
But then there's also smell and all that kind of stuff.
Sure.
I don't know.
The power of imagination or the power of the mechanism
in the human mind that fills the gaps,
that kind of reaches and wants to make the thing you see

in the virtual world real to you.
I believe in that power.
Or humans want to believe.
Yeah.
Like, what if you're lonely?
What if you're sad?
What if you're really struggling in life?
And here's a world where you don't have to struggle anymore.
Humans want to believe so much
that people think the large language models are conscious.
That's how much humans want to believe.
Strong words.
He's throwing left and right hooks.
Why do you think large language models are not conscious?
I don't think I'm conscious.
Oh, so what is consciousness then, George Haas?
It's like what it seems to mean to people.
It's just like a word that atheists use for souls.
Sure.
But that doesn't mean soul is not an interesting word.
If consciousness is a spectrum,
I'm definitely way more conscious
than the large language models are.
I think the large language models
are less conscious than the chicken.
When is the last time you've seen a chicken?
In Miami, like a couple months ago.
How?
No, like a living chicken.
Living chickens walking around Miami, it's crazy.
Like on the street?
Yeah.
Like a chicken.
A chicken, yeah.
All right.
All right, I was trying to call you all
like a good journalist and I got shut down.
Okay, but you don't think much about this kind of
subjective feeling that it feels like something to exist.
And then as an observer, you can have a sense
that an entity is not only intelligent,
but has a kind of subjective experience of its reality.
Like a self-awareness that is capable of like suffering,

of hurting, of being excited by the environment
in a way that's not merely kind of an artificial response,
but a deeply felt one.
Humans wanna believe so much that if I took a rock
and a Sharpie and drew a sad face on the rock,
they'd think the rock is sad.
Yeah.
And you're saying when we look in the mirror,
we apply the same smiley face with rock.
Pretty much, yeah.
Doesn't it?
Isn't that weird though that you're not conscious?
Is that?
No.
But you do believe in consciousness.
Not really.
It's just unclear.
Okay, so to you it's like a little like a symptom
of the bigger thing that's not that important.
Yeah, I mean it's interesting that like the human system
seem to claim that they're conscious.
And I guess it kind of like says something
in a straight up like, okay, what do people mean
when, even if you don't believe in consciousness,
what do people mean when they say consciousness?
And there's definitely like meanings to it.
What's your favorite thing to eat?
Pizza.
Cheese pizza, what are the toppings?
I like cheese pizza.
Don't say pineapple.
No, I don't like pineapple.
Okay.
Pepperoni pizza.
As they put any ham on it, oh, that's real bad.
What's the best pizza?
What are we talking about here?
Like, do you like cheap crappy pizza?
Actually, cargo deep dish, cheese pizza.
Oh, that's my favorite.
There you go.
You bite into a deep dish, a cargo deep dish pizza.
And it feels like you were starving.

You haven't eaten for 24 hours.
You just bite in and you're hanging out
with somebody that matters a lot to you
and you're there with the pizza.
Sounds real nice.
Yeah, all right.
It feels like something.
I'm George motherfucking hot, eating a fucking
Chicago deep dish pizza.
There's just the full peak living experience
of being human, the top of the human condition.
It feels like something to experience that.
Why does it feel like something?
That's consciousness, isn't it?
If that's the word you wanna use to describe it, sure.
I'm not gonna deny that feeling exists.
I'm not gonna deny that I experienced that feeling.
I guess what I kind of take issue to
is that there's some, like,
how does it feel to be a web server?
Do 404s hurt?
Not yet.
How would you know what suffering looked like?
Sure, you can recognize a suffering dog
because we're the same stack as the dog.
All the biostack stuff kind of, especially mammals,
you know, it's really easy, you can.
Game recognizes game.
Yeah, versus the silicon stack stuff.
It's like, you have no idea.
Wow, the little thing has learned to mimic, you know.
But then I realized that that's all we are, too.
Oh, look, the little thing has learned to mimic.
Yeah, I guess, yeah, 404 could be suffering,
but it's so far from our kind of living organism,
our kind of stack, but it feels like AI can start
maybe mimicking the biological stack,
but about it, because it's trained.
Retrained it, yeah.
And so, in that, maybe that's the definition
of consciousness, is the biostat consciousness.
The definition of consciousness
is how close something looks to human.

Sure, I'll give you that one.
No, how close something is to the human experience.
Sure, it's a very anthropocentric definition, but.
Well, that's all we got.
Sure, no, and I don't mean to like,
I think there's a lot of value in it.
Look, I just started my second company,
my third company will be AI Girlfriends.
No, like, I mean.
I wanna find out what your fourth company is after that.
Oh, wow.
Because I think once you have AI Girlfriends,
it's, oh boy, does it get interesting.
Well, maybe let's go there.
I mean, the relationships with AI,
that's creating human-like organisms, right?
And part of being human is being conscious,
is having the capacity to suffer,
having the capacity to experience this life richly
in such a way that you can empathize,
that AI is gonna empathize with you
and you can empathize with it.
Or you can project your anthropomorphic sense
of what the other entity is experiencing.
And an AI model would need to,
yeah, to create that experience inside your mind.
And it doesn't seem that difficult.
Yeah, but, okay, so here's where it actually
gets totally different, right?
When you interact with another human,
you can make some assumptions.
When you interact with these models, you can't.
You can make some assumptions that that other human
experiences suffering and pleasure
in a pretty similar way to you do.
The golden rule applies.
With an AI model, this isn't really true, right?
These large language models are good at fooling people
because they were trained on a whole bunch of human data
and told to mimic it.
Yeah, but if the AI system says,
hi, my name is Samantha,
it has a backstory, I went to college here and there.

Maybe you'll integrate this in the AI system.
I made some chatbots, I gave them backstories,
it was lots of fun.
I was so happy when Lama came out.
Yeah, we'll talk about Lama, we'll talk about all that,
but like, the rock with the smiley face.
Yeah.
Why, it seems pretty natural for you to anthropomorphize
that thing and then start dating it.
And before you know it, you're married and have kids.
Who's a rock?
Who's a rock?
There's pictures on Instagram with you and a rock
and a smiley face.
To be fair, like, you know, something that people
generally look for when they're looking for someone
to date is intelligence in some form.
And the rock doesn't really have intelligence,
only a pretty desperate person would date a rock.
I think we're all desperate deep down.
Oh, not rock-level desperate.
All right.
Not rock-level desperate, but AI-level desperate.
I don't know, I think all of us have a deep loneliness.
It just feels like the language models are there.
Oh, I agree.
And you know what, I won't even say this so cynically.
I will actually say this in a way that like,
I want AI friends, I do.
Yeah.
Like I would love to, you know, again,
the language models now are still a little,
like people are impressed with these GPT things
and I look at like, or like, or the copilot, the coding one.
And I'm like, okay, this is like junior engineer level
and these people are like Fiverr-level artists
and copywriters.
Like, okay, great.
We got like Fiverr and like junior engineers.
Okay, cool.
Like, and this is just the start
and it will get better, right?
Like I can't wait to have AI friends

who are more intelligent than I am.
So Fiverr is just a temporary, it's not the ceiling.
No, definitely not.
Is it, is the count as cheating
when you're talking to an AI model, emotional cheating?
That's, that's up to you
when you're human partner to define.
Oh, you have to, all right.
You're getting, yeah.
You have to have that conversation, I guess.
All right.
I mean, integrate that with, with porn and all this stuff.
No, I mean, it's similar kind of to porn.
Yeah.
Yeah.
Right, I think people in relationships
have different views on that.
Yeah, but most people don't have like serious,
open conversations about all the different aspects
of what's cool and what's not.
And it feels like AI is a really weird conversation to have.
The porn one is a good branching off, like these things.
You know, one of my scenarios that I put in my chat bot
is a, you know, a nice girl named Lexi.
She's 20.
She just moved out to LA.
She wanted to be an actress,
but she started doing OnlyFans instead
and you're on a date with her.
Enjoy.
Oh man.
Yeah.
And so is that if you're actually dating somebody
in real life, is that cheating?
I feel like it gets a little weird.
Sure.
It gets real weird.
It's like, what are you allowed to say to an AI bot?
Imagine having that conversation with a significant other.
I mean, these are all things from people to define
in their relationships.
What it means to be human is just gonna start to get weird.
Especially online.

Like, how do you know?
Like, there'll be moments when you'll have
what you think is a real human you interacted with
on Twitter for years and you realize it's not.
I spread, I love this meme, heaven banning.
Do you know what shadow banning?
Yeah.
All right, shadow banning.
Okay, you post, no one can see it.
Heaven banning, you post, no one can see it,
but a whole lot of AIs are spot up to interact with you.
Well, maybe that's what the way human civilization ends
is all of us are heaven banned.
There's a great, it's called My Little Pony friendship
is optimal.
It's a sci-fi story that explores this idea.
Friendship is optimal.
Friendship is optimal.
Yeah, I'd like to have some, at least stuff
on the intellectual realm, some AI friends
that argue with me, but the romantic realm is weird.
Definitely weird, but not out of the realm of
the kind of weirdness that human civilization
is capable of, I think.
I want it, look, I want it.
If no one else wants it, I want it.
Yeah, I think a lot of people probably want it.
There's a deep loneliness.
And I'll feel their loneliness,
and it just will only advertise to you some of the time.
Yeah, maybe the conceptions of monogamy changed too.
Like I grew up in a time, like I value monogamy,
but maybe that's a silly notion
when you have arbitrary number of AI systems.
This interesting path from rationality to polyamory,
that doesn't make sense for me.
For you, but you're just a biological organism
that was born before the internet really took off.
The crazy thing is, culture is whatever we define it as.
These things are not, is a lot problem
in moral philosophy, right?
There's no like, okay, what is might be
that like computers are capable of mimicking

girlfriends perfectly.
They passed the girlfriend Turing test, right?
But that doesn't say anything about a lot.
That doesn't say anything about how we ought
to respond to them as a civilization.
That doesn't say we ought to get rid of monogamy, right?
That's a completely separate question,
really a religious one.
Girlfriend Turing test, I wonder what that looks like.
Girlfriend Turing test.
Are you writing that?
Will you be the Alan Turing of the 21st century
that writes the Girlfriend Turing test?
No, I mean, of course my AI girlfriends,
their goal is to pass the Girlfriend Turing test.
No, but there should be like a paper
that kind of defines the test.
I mean, the question is if it's deeply personalized
or there's a common thing that really gets everybody.
Yeah, I mean, you know, look, we're a company,
we don't have to get everybody,
we just have to get a large enough clientele to stay with us.
I like how you're already thinking company.
All right, let's, before we go to company number three
and company number four, let's go to company number two.
All right.
Tinycorp, possibly one of the greatest names
of all time for a company.
You've launched a new company called Tinycorp
that leads the development of Tinygrad.
What's the origin story of Tinycorp and Tinygrad?
I started Tinygrad as like a toy project
just to teach myself, okay, like what is a convolution?
What are all these options you can pass to them?
What is the derivative of a convolution, right?
Very similar to a Carpathian micrograd, very similar.
And then I started realizing,
I started thinking about like AI chips.
I started thinking about chips that run AI
and I was like, well, okay,
this is going to be a really big problem.
If NVIDIA becomes a monopoly here,
how long before NVIDIA is nationalized?

So you, one of the reasons that start Tinycorp
is to challenge NVIDIA?
It's not so much to challenge NVIDIA.
I actually, I like NVIDIA and it's,
to make sure power stays decentralized.
Yeah.
And here's computational power.
I see you NVIDIA is kind of locking down
the computational power of the world.
If NVIDIA becomes just like 10x better than everything else,
you're giving a big advantage to somebody
who can secure NVIDIA as a resource.
Yeah.
In fact, if Jensen watches this podcast,
he may want to consider this.
He may want to consider making sure
his company is not nationalized.
Do you think that's an actual threat?
Oh, yes.
No, but there's so much, you know, there's AMD.
So we have NVIDIA and AMD, great.
All right.
But you don't think there's like a push
towards like selling, like Google selling TPUs
or something like this.
You don't think there's a push for that?
Have you seen it?
Google loves to rent you TPUs.
It doesn't, you can't buy it at Best Buy?
Yeah.
So I started work on a chip.
I was like, okay, what's it gonna take to make a chip?
And my first notions were all completely wrong
about why, about like how you could improve on GPUs.
And I will take this, this is from Jim Keller
on your podcast.
And this is one of my absolute favorite
descriptions of computation.
So there's three kinds of computation paradigms
that are common in the world today.
There's CPUs and CPUs can do everything.
CPUs can do add and multiply.
They can do load and store

and they can do compare and branch.
And when I say they can do these things,
they can do them all fast, right?
So compare and branch are unique to CPUs.
And what I mean by they can do them fast
is they can do things like branch prediction
and speculative execution.
And they spend tons of transistors
and they use like super deep reorder buffers
in order to make these things fast.
Then you have a simpler computation model GPUs.
GPUs can't really do compare and branch.
I mean they can, but it's horrendously slow.
But GPUs can do arbitrary load and store, right?
GPUs can do things like X dereference Y.
So they can fetch from arbitrary pieces of memory.
They can fetch from memory that is defined
by the contents of the data.
The third model of computation is DSPs.
And DSPs are just add and multiply, right?
Like they can do load and stores
but only static load and stores.
Only loads and stores that are known
before the program runs.
And you look at neural networks today
and 95% of neural networks are all the DSP paradigm.
They are just statically scheduled ads and multiplies.
So TinyGuard really took this idea
and I'm still working on it to extend this
as far as possible.
Every stage of the stack has turn completeness, right?
Python has turn completeness.
And then we take Python, we go to C++ which is turn complete.
And maybe C++ calls into some CUDA kernels
which are turn complete.
The CUDA kernels go through LVM which is turn complete
into PTX which is turn complete,
to SAS which is turn complete on a turn complete processor.
I wanna get turn completeness out of the stack entirely.
Because once you get rid of turn completeness
you can reason about things.
Rice's theorem and the halting problem
do not apply to AdMol machines.

Okay, what's the power and the value
of getting turn completeness out of?
Are we talking about the hardware or the software?
Every layer of the stack.
Every layer.
Every layer of the stack removing turn completeness
allows you to reason about things, right?
So the reason you need to do branch prediction at a CPU
and the reason it's prediction
and the branch predictors are,
I think they're like 99% on CPUs.
Why did they get 1% of them wrong?
Well, they get 1% wrong because you can't know, right?
That's the halting problem.
It's equivalent to the halting problem
to say whether a branch is gonna be taken or not.
I can show that, but the AdMol machine,
the neural network runs the identical compute every time.
The only thing that changes is the data.
So when you realize this, you think about,
okay, how can we build a computer
and how can we build a stack
that takes maximal advantage of this idea?
So what makes TinyGrad different
from other neural network libraries
is it does not have a primitive operator
even for matrix multiplication.
And this is every single one.
They even have primitive operators
and there's things like convolutions.
So no MatMol.
No MatMol.
Well, here's what a MatMol is.
So I'll use my hands to talk here.
So if you think about a cube
and I put my two matrices that I'm multiplying
on two faces of the cube, right?
You can think about the matrix multiply as, okay,
the n cubed, I'm gonna multiply for each one in the cubed.
And then I'm gonna do a sum, which is a reduce
up to here to the third face of the cube
and that's your multiplied matrix.
So what a matrix multiply is,

is a bunch of shape operations, right?
A bunch of permute three shapes
and expands on the two matrices.
Multiply and cubed, a reduce and cubed,
which gives you an n squared matrix.
Okay, so what is the minimum number of operations
that can accomplish that
if you don't have MatMol as a primitive?
So tiny grad has about 20.
And you can compare tiny grads,
offset or IR to things like XLA or PrimTorch.
So XLA and PrimTorch are ideas where like,
okay, Torch has like 2,000 different kernels.
PyTorch 2.0 introduced PrimTorch, which has only 250.
Tiny grad has order of magnitude 25.
It's 10x less than XLA or PrimTorch.
And you can think about it as kind of like risk versus sysc,
right?
These other things are sysc like systems.
Tiny grad is risk.
And risk one.
Risk architecture is gonna change everything.
1995, hackers.
Wait, really? That's an actual thing?
Angelina Jolie delivers the line.
Risk architecture is gonna change everything in 1995.
And here we are with arm in the phones
and arm everywhere.
Wow, I love it when movies actually have real things in them.
Right.
Okay, interesting.
So this is like,
so you're thinking of this as the risk architecture
of ML stack.
25, what, can you go through the
the four op types?
Sure.
Okay, so you have unary ops,
which take in a tensor and return a tensor of the same size
and do some unary op to it.
X log, reciprocal, sign, right?
They take in one and they're point-wise.
Relu.

Yeah, relu.
Almost all activation functions are unary ops.
Some combinations of unary ops together is still unary op.
Then you have binary ops.
Binary ops are like point-wise addition,
multiplication, division, compare.
It takes in two tensors of equal size
and outputs one tensor.
Then you have reduce ops.
Reduce ops will like take a three-dimensional tensor
and turn it into a two-dimensional tensor
or a three-dimensional tensor
turn it into zero-dimensional tensor.
Things like a sum or a max
are really the common ones there.
And then the fourth type is movement ops.
And movement ops are different from the other types
because they don't actually require computation.
They require different ways to look at memory.
So that includes reshapes, permutes, expands, flips.
Those are the main ones, probably.
And so with that, you have enough to make a map mall.
And convolutions.
And every convolution you can imagine,
dilated convolutions, strided convolutions,
transposed convolutions.
You're right on GitHub about laziness.
Showing a map mall, matrix multiplication.
See how despite the style,
it is fused into one kernel with the power of laziness.
Can you elaborate on this power of laziness?
Sure.
So if you type in PyTorch A times B plus C,
what this is going to do is it's going to first multiply
add and B, A and B, and store that result into memory.
And then it is going to add C by reading that result
from memory, reading C from memory
and writing that out to memory.
There is way more loads and stores to memory
than you need there.
If you don't actually do A times B, as soon as you see it,
if you wait until the user actually realizes that tensor,
until the laziness actually resolves,

you confuse that plus C.
This is like, it's the same way Haskell works.
So what's the process of porting a model into TinyGrad?
So TinyGrad's front end looks very similar to PyTorch.
I probably could make a perfect,
or pretty close to perfect interop layer
if I really wanted to.
I think that there's some things that are nicer
about TinyGrad syntax than PyTorch,
but the front end looks very torch-like.
You can also load in Onyx models.
We have more Onyx tests passing than Core ML.
Core ML.
Okay, so...
We'll pass Onyx around time soon.
What about like the developer experience with TinyGrad?
What it feels like, versus PyTorch?
By the way, I really like PyTorch.
I think that it's actually a very good piece of software.
I think that they've made a few different trade-offs,
and these different trade-offs are where, you know,
TinyGrad takes a different path.
One of the biggest differences is it's really easy to see
the kernels that are actually being sent to the GPU.
Right?
If you run PyTorch on the GPU,
you like do some operation,
and you don't know what kernels ran.
You don't know how many kernels ran.
You don't know how many flops were used.
You don't know how much memory accesses were used.
TinyGrad type debug equals two,
and it will show you in this beautiful style
every kernel that's run.
How many flops, and how many bytes.
So can you just linger on what problem TinyGrad solves?
TinyGrad solves the problem
of porting new ML accelerators quickly.
One of the reasons, tons of these companies now,
I think Sequoia marked Graphcore to zero, right?
Seribus, TensTorrent, Grock,
all of these ML accelerator companies, they built chips.
The chips were good.

The software was terrible.
And part of the reason is
because I think the same problem is happening with Dojo.
It's really, really hard to write a PyTorch port
because you have to write 250 kernels
and you have to tune them all for performance.
What does Jim Color think about TinyGrad?
You guys hung on quite a bit.
So he's, he was involved with TensTorrent.
What's his praise and what's his criticism
of what you're doing with your life?
Look, my prediction for TensTorrent
is that they're gonna pivot to making risk five chips.
CPUs.
CPUs, why?
Well, because AI accelerators are a software problem,
not really a hardware problem.
Oh, interesting.
So you don't think, you think the diversity of AI accelerators
in the hardware space is not going to be a thing
that exists long-term?
I think what's gonna happen is if I can fit it, okay.
If you're trying to make an AI accelerator,
you better have the capability of writing
a Torch level performance stack on NVIDIA GPUs.
If you can't write a Torch stack on NVIDIA GPUs,
and I mean all the way, I mean down to the driver,
there's no way you're gonna be able to write it on your chip
because your chip's worse than an NVIDIA GPU.
The first version of the chip you tape out,
it's definitely worse.
Oh, you're saying writing that stack is really tough?
Yes.
And not only that, actually the chip that you tape out
almost always because you're trying to get advantage
over NVIDIA, you're specializing the hardware more.
It's always harder to write software
for more specialized hardware.
Like a GPU is pretty generic.
And if you can't write an NVIDIA stack,
there's no way you can write a stack for your chip.
So my approach with TinyGrad is first,
write a performance NVIDIA stack, or we're targeting AMD.

So you did say a few to NVIDIA a little bit, with love.
With love.
Yeah, with love.
It's like the Yankees, you know, I'm a Mets fan.
Oh, you're a Mets fan, a risk fan and a Mets fan.
What's the hope that AMD has?
You did build with AMD recently that I saw.
How does the 7900 XTX compare to the RTX 4090 or 4080?
Well, let's start with the fact
that the 7900 XTX kernel drivers don't work.
And if you run demo apps in loops, it panics the kernel.
Okay, so this is a software issue.
Lisa Sue responded to my email.
Oh.
I reached out, I was like, this is, you know, really?
Like, I understand if you're seven by seven
transposed Winograd comm is slower than NVIDIAs,
but literally when I run demo apps in a loop,
the kernel panics.
So just adding that loop.
Yeah, I just literally took their demo apps
and wrote like, while true, semicolon do the app,
semicolon done in a bunch of screens, right?
This is like the most primitive fuzz testing.
Why do you think that is?
They're just not seeing a market in machine learning?
They're trying to change.
They're trying to change.
And I had a pretty positive interaction with them this week.
Last week, I went on YouTube, I was just like,
that's it, I give up on AMD.
Like this is the, their driver doesn't even like,
I'm not gonna, I'm not gonna, you know,
I'll go with Intel GPUs, right?
Intel GPUs have better drivers.
So you're kind of spearheading
the diversification of GPUs.
Yeah, and I'd like to extend that diversification
to everything.
I'd like to diversify the, right?
The more my central thesis about the world is,
there's things that centralize power and they're bad.
And there's things that decentralize power and they're good.

Everything I can do to help decentralize power,
I'd like to do.
So you're really worried about the centralization
of Nvidia, that's interesting.
And you don't have a fundamental hope
for the proliferation of ASICs, except in the cloud.
I'd like to help them with software.
No, actually there's only, the only ASIC
that is remotely successful is Google's TPU.
And the only reason that's successful is because Google
wrote a machine learning framework.
I think that you have to write a competitive
machine learning framework in order to be able
to build an ASIC.
You think meta with PyTorch builds a competitor?
I hope so.
They have one, they have an internal one.
Internal, I mean, public facing with a nice cloud
interface and so on.
I don't want a cloud.
You don't like cloud?
I don't like cloud.
What do you think is the fundamental limitation of cloud?
Fundamental limitation of cloud is who owns the off switch.
So it's the power to the people.
Yeah.
And you don't like the man to have all the power.
Exactly.
All right.
And right now the only way to do that is with the NVIDIA GPUs
if you want performance and stability.
Interesting, it's a costly investment emotionally
to go with AMDs.
Well, let me sort of on a tangent ask you,
what, you've built quite a few PCs.
What's your advice on how to build a good custom PC
for let's say for the different applications that you use
for gaming, for machine learning?
Well, you shouldn't build one.
You should buy a box from the TinyCorp.
I heard rumors, whispers about this box in the TinyCorp.
What's this thing look like?
What is it?

What is it called?
It's called the TinyBox.
TinyBox?
It's $15,000.
And it's almost a paid-a-flop of compute.
It's over 100 gigabytes of GPU RAM.
It's over five terabytes per second of GPU memory bandwidth.
I'm gonna put like four NVMEs in RAID.
You're gonna get like 20, 30 gigabytes per second
of drive read bandwidth.
I'm gonna build like the best deep learning box
that I can that plugs into one wall outlet.
Okay, can you go through those specs again
a little bit from memory?
Yeah, so it's almost a paid-a-flop of compute.
So MD and TAL.
Today, I'm leaning toward AMD.
But we're pretty agnostic to the type of compute.
The main limiting spec is a 120 volt 15 amp circuit.
Okay.
Well, I mean it, because in order to like,
there's a plug over there, all right?
You have to be able to plug it in.
We're also gonna sell the tiny rack,
which like, what's the most power you can get into your house
without arousing suspicion?
And one of the answers is an electric car charger.
Wait, where does the rack go?
Your garage.
Interesting, the car charger.
A wall outlet is about 1,500 watts.
A car charger is about 10,000 watts.
What is the most amount of power
you can get your hands on without arousing suspicion?
George Haas.
Okay.
So the tiny box and you said NVMEs and RAID.
I forget what you said about memory,
all that kind of stuff.
Okay, what about what GPUs?
Again, probably 7,900 XTXs, but maybe 3090s,
maybe a 770s, if anyone tells.
You're flexible or still exploring?

I'm still exploring.
I wanna deliver a really good experience to people.
And yeah, what GPUs I end up going with?
Again, I'm leaning toward AMD.
It will, we'll see.
You know, in my email, what I said to AMD is like,
just dumping the code on GitHub is not open source.
Open source is a culture.
Open source means that your issues
are not all one year old stale issues.
Open source means developing in public.
And if you guys can commit to that,
I see a real future for AMD as a competitor to video.
Well, I'd love to get a tiny box at MIT.
So whenever it's ready, let's do it.
We're taking pre-orders.
I took this from Elon.
I'm like $100 fully refundable pre-orders.
Is it gonna be like the Cybertruck?
It's gonna take a few years or?
No, I'll try to do it faster.
It's a lot simpler than a truck.
Well, there's complexities,
not to just the putting the thing together,
but like shipping and all this kind of stuff.
The thing that I wanna deliver to people out of the box
is being able to run 65 billion parameter Lama
in FP16 in real time, in like a good,
like 10 tokens per second
or five tokens per second or something.
Just, it works.
Lama's running or something like Lama.
Experient, yeah, or I think Falcon is the new one.
Experience a chat with the largest language model
that you can have in your house.
Yeah, from a wall plug.
From a wall plug, yeah.
Actually, for inference,
it's not like even more power would help you get more.
Even more power wouldn't get you more.
The biggest model released
is 65 billion parameter Lama as far as I know.
So it sounds like TinyBox

will naturally pivot towards company number three,
because you could just get the girlfriend or boyfriend.
That one's harder, actually.
The boyfriend is harder?
The boyfriend's harder, yeah.
I think that's a very biased statement.
I think a lot of people would disagree.
What, why is it harder to replace a boyfriend
than the girlfriend with the artificial LLM?
Because women are attracted to status and power,
and men are attracted to youth and beauty.
No, I mean, that's what I mean.
Both are immeasurable, easy through the language model.
No, no machines do not have any status or real power.
I don't know, I think you both,
well, first of all, you're using language mostly
to communicate youth and beauty and power and status.
But status fundamentally is a zero sum game.
Whereas youth and beauty are not.
No, I think status is a narrative you can construct.
I don't think status is real.
I don't know.
I just think that that's why it's harder.
You know, yeah, maybe it is my biases.
I think status is way easier to fake.
I also think that men are probably more desperate
and more likely to buy my product,
so maybe they're a better target market.
Desperation is interesting.
Easier to fool.
I can see that.
Yeah, look, I mean, look,
I know you can look at porn viewership numbers, right?
A lot more men watch porn than women.
You can ask why that is.
Well, there's a lot of questions and answers
you can get there.
Anyway, with the TinyBox, how many GPUs in TinyBox?
Six.
Six.
Oh, man.
And I'll tell you why it's six.
Yeah.

So AMD Epic processors have 128 lanes of PCIe.
I want to leave enough lanes for some drives
and I want to leave enough lanes for some networking.
How do you do cooling for something like this?
Ah, that's one of the big challenges.
Not only do I want the cooling to be good,
I want it to be quiet.
I want the TinyBox to be able to sit comfortably
in your room, right?
This is really going towards the girlfriend thing.
Cause you want to run the LLM.
I'll give a more, I mean,
I can talk about how it relates to company number one.
Call my AI.
Well, but yes, quiet, oh, quiet
because you may be potentially want to run in a car.
No, no quiet because you want to put this thing
in your house and you want it to coexist with you.
If it's screaming at 60 dB, you don't want that
in your house, you'll kick it out.
60 dB, yeah.
I want like 40, 45.
So how do you make the cooling quiet?
That's an interesting problem in itself.
A key trick is to actually make it big.
Ironically, it's called the TinyBox.
But if I can make it big,
a lot of that noise is generated because of high pressure air.
If you look at like a OneU server,
a OneU server has these super high pressure fans
that like super deep and they're like,
Genesis versus if you have something that's big,
well, I can use a big,
and you know, they call them big ass fans,
those ones that are like huge on the ceiling
and they're completely silent.
So TinyBox will be big.
It is the, I do not want it to be large according to UPS.
I want it to be shippable as a normal package,
but that's my constraint there.
Interesting.
Well, the fans stuff, can't it be assembled on location?
No, no.

No.
No, it has to be, well, you're...
Look, I want to give you a great out of the box experience.
I want you to lift this thing out.
I want it to be like the Mac, you know, TinyBox.
The Apple experience.
Yeah.
I love it.
Okay.
So TinyBox would run TinyGrad.
Like what do you envision this whole thing to look like?
We're talking about like Linux with a full
software engineering environment
and just not PyTorch but TinyGrad.
Yeah.
We did a poll if people want Ubuntu or Arch.
We're going to stick with Ubuntu.
Ooh, interesting.
What's your favorite flavor of Linux?
Ubuntu.
Ubuntu.
I like Ubuntu, Mate.
However, we pronounce that, Mate.
So how do you, you've gotten Llama into TinyGrad.
You've gotten stable diffusion into TinyGrad.
What was that like?
Can you comment on like, what are these models?
What's interesting about porting them?
So...
Yeah, like what are the challenges?
What's naturally, what's easy, all that kind of stuff.
There's a really simple way to get these models
into TinyGrad and you can just export them as Onyx
and then TinyGrad can run Onyx.
So the ports that I did of Llama, stable diffusion
and now Whisper are more academic to teach me
about the models, but they are cleaner
than the PyTorch versions.
You can read the code.
I think the code is easier to read.
It's less lines.
There's just a few things about the way TinyGrad writes
things.

Here's a complaint I have about PyTorch.
nn.relu is a class, right?
So when you create a, when you create an nn module,
you'll put your nnrelus as in a knit.
And this makes no sense.
Nnrelu is completely stateless.
Why should that be a class?
But that's more like a software engineering thing.
Or do you think it has a cost on performance?
Oh no, it doesn't have a cost on performance.
But yeah, no, I think that it's,
that's what I mean about like TinyGrad's front end
being cleaner.
I see.
What do you think about Mojo?
I don't know if you've been paying attention
to the programming language that does some interesting ideas
that kind of intersect TinyGrad.
I think that there is a spectrum.
And like on one side you have Mojo,
and on the other side you have like GGML.
GGML is this like, we're gonna run Lama fast on Mac.
And okay, we're gonna expand out to a little bit,
but we're gonna basically go to like depth first, right?
Mojo is like, we're gonna go breadth first.
We're gonna go so wide that we're gonna make
all of Python fast and TinyGrad's in the middle.
TinyGrad is, we are going to make neural networks fast.
Yeah, but they try to really get it to be fast,
compiled down to the specifics hardware,
and make that compilation step as flexible
and resilient as possible.
Yeah, but they have turn completeness.
And that limits you.
Turn?
That's what you're seeing, it's somewhere in the middle.
So you're actually going to be targeting some accelerators,
some, like some number, not one.
My goal is step one,
build an equally performance stack to PyTorch
on NVIDIA and AMD, but with way less lines.
And then step two is, okay, how do we make an accelerator,
right, but you need step one.

You have to first build the framework
before you can build the accelerator.
Can you explain ML Perf?
What's your approach in general to benchmark
and TinyGrad performance?
So I'm much more of a, like build it the right way
and worry about performance later.
There's a bunch of things where I haven't even like
really dove into performance.
The only place where TinyGrad is competitive performance wise
right now is on Qualcomm GPUs.
So TinyGrad is actually used in OpenPilot to run the model.
So the driving model is TinyGrad.
When did that happen, that transition?
About eight months ago now.
And it's two X faster than Qualcomm's library.
What's the hardware that OpenPilot runs on,
the Comma Air?
It's a Snapdragon 845.
Okay.
So this is using the GPU.
So the GPU is an Adreno GPU.
There's like different things.
There's a really good Microsoft paper
that talks about like mobile GPUs
and why they're different from desktop GPUs.
One of the big things is in a desktop GPU,
you can use buffers on a mobile GPU image textures
a lot faster.
On a mobile GPU image textures and limit, okay.
And so you want to be able to leverage that.
I want to be able to leverage it
in a way that it's completely generic, right?
So there's a lot of this.
Xiaomi has a pretty good open source library
from old GPUs called Mace
where they can generate where they have these kernels,
but they're all hand coded, right?
So that's great if you're doing three by three comps.
That's great if you're doing dense map models,
but the minute you go off the beaten path at TinyBit,
well, your performance is nothing.
Since you mentioned OpenPilot,

I'd love to get an update in the company number one,
Kama AI world.
How are things going there in the development
of semi-autonomous driving?
You know, almost no one talks about FSD anymore
and even less people talk about OpenPilot.
We've solved the problem.
Like we solved it years ago.
What's the problem exactly?
Well, how do you-
What is solving it mean?
Solving means how do you build a model
that outputs a human policy for driving?
How do you build a model that given
a reasonable set of sensors
outputs a human policy for driving?
So you have companies like Waymoon Cruise
which are hand coding these things
that are like quasi-human policies.
Then you have Tesla and maybe even to more
of an extent, Kama asking,
okay, how do we just learn human policy from data?
The big thing that we're doing now
and we just put it out on Twitter,
at the beginning of Kama,
we published a paper called Learning a Driving Simulator.
And the way this thing worked was it's a,
it was an auto encoder and then an RNN in the middle.
Right?
You take an auto encoder, you compress the picture,
you use an RNN, predict the next state
and these things were, you know,
it was a laughably bad simulator.
Right, this is 2015 error machine learning technology.
Today, we have VQ, VAE and Transformers.
We're building Drive GPT basically.
Drive GPT.
Okay, so, and it's trained on what?
Is it trained in a self-supervised way?
It's trained on all the driving data
to predict the next frame.
So really trying to learn a human policy.
What would a human do?

Well, actually our simulator is conditioned on the pose.
So it's actually a simulator.
You can put in like a state action pair
and get up the next state.
Okay.
And then once you have a simulator,
you can do RL in the simulator
and RL will get us that human policy.
So it transfers.
Yay.
RL with a reward function.
Not asking, is this close to the human policy
but asking, what a human disengage
if you did this behavior?
Okay, let me think about the distinction there.
What a human disengage.
What a human disengage.
That correlates, I guess, with the human policy
but it can be different.
So it doesn't just say, what would a human do?
It says, what would a good human driver do?
And such that the experience is comfortable
but also not annoying in that like,
the thing is very cautious.
So it's finding a nice balance.
That's interesting.
It's nice.
It's asking exactly the right question.
What will make our customers happy?
Right.
That you never want to disengage.
Because usually disengagement is almost always a sign
of I'm not happy with what the system is doing.
Usually, there's some that are just,
I felt like driving and those are always fine too
but they're just going to look like noise in the data.
But even I felt like driving.
Maybe, yeah.
That's even that's a signal like,
why do you feel like driving here?
You need to recalibrate your relationship with the car.
Okay, so that's really interesting.
How close are we to solving self-driving?

It's hard to say.
We haven't completely closed the loop yet.
So we don't have anything built
that truly looks like that architecture yet.
We have prototypes and there's bugs.
So we are, a couple of bug fixes away.
Might take a year, might take 10.
What's the nature of the bugs?
Are these major philosophical bugs, logical bugs?
What kind of bugs are we talking about?
They're just like, they're just like stupid bugs.
And like also we might just need more scale.
We just massively expanded our compute cluster at comma.
We now have about two people worth of compute,
40 petaflops.
Well, people are different.
20 petaflops, that's a person.
It's just a unit, right?
Horses are different too
but we still call it a horsepower.
Yeah, but there's something different about mobility
than there is about perception and action
in a very complicated world.
But yes.
Well, yeah, of course.
Not all flops are created equal.
If you have randomly initialized weights, it's not gonna.
Not all flops are created equal.
So flops are doing way more useful things than others.
Yep, yep.
Tell me about it.
Okay, so more data.
Scale means more scale in compute
or scale in scale of data?
Both.
Diversity of data?
Diversity is very important in data.
Yeah, I mean, we have,
so we have about, I think we have like 5,000 daily actives.
How would you evaluate how FSD is doing?
Pretty well.
Stop driving.
Pretty well.

How's that race going between comma AI and FSD?
Tesla has always wanted two years ahead of us.
They've always been one to two years ahead of us
and they probably always will be
because they're not doing anything wrong.
What have you seen that since the last time we talked,
they're interesting architectural decisions,
training decisions,
like the way they deploy stuff,
the architectures they're using in terms of the software,
how the teams are run, all that kind of stuff,
data collection, anything interesting?
I mean, I know they're moving toward more of an end-to-end approach.
So creeping towards end-to-end as much as possible
across the whole thing,
the training, the data collection, everything.
They also have a very fancy simulator.
They're probably saying all the same things we are.
They're probably saying we just need to optimize,
you know, what is the reward?
We'll get negative reward for disengagement, right?
Like, everyone kind of knows this.
It's just a question of who can actually build
and deploy the system.
Yeah, I mean, this requires good software engineering,
I think. Yeah.
And the right kind of hardware.
Yeah, and hardware to run it.
You still don't believe in cloud in that regard?
I have a compute cluster in my, oh, 800 amps.
Tiny grad.
It's 40 kilowatts at idle, our data center.
That's crazy.
We have 40 kilowatts just burning
just when the computers are idle.
Oh, sorry, sorry, compute cluster.
Compute cluster, I got it.
It's not a data center.
Yeah, yeah.
Now, data centers are clouds.
We don't have clouds.
Data centers have air conditioners.
We have fans.

That makes it a compute cluster.
I'm guessing this is a kind of a legal distinction
that's about to make. Sure, yeah.
We have a compute cluster.
You said that you don't think LLMs have consciousness,
or at least not more than a chicken.
Do you think they can reason?
Is there something interesting to you
about the word reason,
about some of the capabilities
that we think is kind of human,
to be able to integrate complicated information
and through a chain of thought,
arrive at a conclusion that feels novel.
A novel integration of disparate facts.
Yeah, I don't think that there's,
I think that I can reason better than a lot of people.
Hey, isn't that amazing to you though?
Isn't that like an incredible thing
that a transformer can achieve?
I mean, I think that calculators can add better
than a lot of people.
But language feels like reasoning
through the process of language,
which looks a lot like thought.
Making brilliancies in chess,
which feels a lot like thought.
Whatever new thing that AI can do,
everybody thinks is brilliant.
And then like 20 years go by and they're like,
well, you have a chess, that's like mechanical.
Like adding, that's like mechanical.
So you think language is not that special?
It's like chess.
It's like chess and it's like.
I don't know, because it's very human.
We take it, listen, there's something different
between chess and language.
Chess is a game that a subset of population plays.
Language is something we use nonstop
for all of our human interaction.
And human interaction is fundamental to society.
So it's like, holy shit,

this language thing is not so difficult
to like create in a machine.
The problem is if you go back to 1960
and you tell them that you have a machine
that can play amazing chess,
of course someone in 1960 will tell you
that machine is intelligent.
Someone in 2010 won't, what's changed, right?
Today we think that these machines
that have language are intelligent,
but I think in 20 years we're gonna be like,
yeah, but can it reproduce?
So reproduction, yeah, we may redefine
what it means to be, what is it?
A high performance living organism on earth.
Humans are always gonna define a niche for themselves.
Like, well, you know, we're better than the machines
because we can, you know,
and like they tried creative for a bit,
but no one believes that one anymore.
But niche is that delusional
or is there some accuracy to that?
Because maybe like with chess,
you start to realize like that we have
ill-conceived notions of what makes humans special.
Like the apex organism on earth.
Yeah, and I think maybe we're gonna go through
that same thing with language.
And that same thing with creativity.
But language carries these notions of truth and so on.
And so we might be like, wait,
maybe truth is not carried by language.
Maybe there's like a deeper thing.
The niche is getting smaller.
Oh boy.
But no, no, no, you don't understand.
Humans are created by God
and machines are created by humans.
Therefore, right?
Like that'll be the last niche we have.
So what do you think about this,
the rapid development of LMS?
If we could just like stick on that,

it's still incredibly impressive.
Like with ChagYBT, just even ChagYBT,
what are your thoughts about reinforcement learning
with human feedback on these large language models?
I'd like to go back to when calculators first came out
and or computers.
And like, I wasn't around.
Look, I'm 33 years old.
And to like see how that affected
like society.
Maybe you're right.
So I wanna put on the big picture hat here.
Oh my God, refrigerator, wow.
The refrigerator, electricity, all that kind of stuff.
But you know, with the internet,
large language models seeming human
like basically passing a touring test.
It seems it might have really at scale,
rapid transformative effects on society.
But you're saying like other technologies have as well.
So maybe calculator's not the best example of that
because that just seems like a may,
well, no, maybe calculator.
But the poor milk man,
the day he learned about refrigerators,
he's like, I'm done.
You tell me, you can just keep the milk in your house.
You don't even need to deliver it every day, I'm done.
Well, yeah, you have to actually look
at the practical impacts of certain technologies
that they've had.
Yeah, probably electricity is a big one.
And also how rapidly it's spread.
Man, the internet's a big one.
I do think it's different this time though.
Yeah, it just feels like stuff.
The niche is getting smaller.
The niche that humans, that makes humans special.
It feels like it's getting smaller rapidly though,
doesn't it?
Or is that just the feeling we dramatize everything?
I think we dramatize everything.
I think that you ask the milk man

when he saw our refrigerators,
and they're gonna have one of these in every home?
Yeah, yeah, yeah.
Yeah, but boys are impressive.
So much more impressive than seeing
a chess world champion AI system.
I disagree actually.
I disagree.
I think things like Mu Zero and Alpha Go
are so much more impressive,
because these things are playing
beyond the highest human level.
The language models are writing middle school level essays,
and people are like, wow, it's a great essay.
It's a great five paragraph essay
about the causes of the Civil War.
Okay, forget the Civil War, just generating code, codex.
Oh!
So you're saying it's mediocre code.
Terrible.
I don't think it's terrible.
I think it's just mediocre code.
Often close to correct.
Like for mediocre purposes.
That's the scariest kind of code.
I spent 5% of time typing and 95% of time debugging.
The last thing I want is close to correct code.
I want a machine that can help me with the debugging,
not with the typing.
You know, it's like L2, level two driving,
similar kind of thing.
Yeah, you still should be a good programmer
in order to modify.
I wouldn't even say debugging.
Just modifying the code, reading it.
Don't think it's like level two driving.
I think driving is not tool complete, and programming is.
Meaning you don't use the best possible tools to drive.
You're not like, cars have basically the same interface
for the last 50 years.
Computers have a radically different interface.
Okay, can you describe the concept of tool complete?
So think about the difference between a car

from 1980 and a car from today.
No difference really.
It's got a bunch of pedals, it's got a steering wheel.
Great.
Maybe now it has a few ADAS features,
but it's pretty much the same car.
You have no problem getting into a 1980 car and driving it.
You take a programmer today
who spent their whole life doing JavaScript,
and you put him in an Apple 2e prompt,
and you tell him about the line numbers in basic.
But how do I insert something between line 17 and 18?
Oh, wow.
But so in tool, you're putting in the programming languages.
So it's just the entire stack of the tooling.
So it's not just like the IDEs or something like this,
it's everything.
Yes, it's IDEs, the languages, the runtimes,
it's everything, and programming is tool complete.
So almost if Codex or Copilot are helping you,
that actually probably means
that your framework or library is bad,
and there's too much boilerplate in it.
Yeah, but don't you think
so much programming has boilerplate?
Tinygrad is now 2,700 lines,
and it can run Lama and stable diffusion.
And all of this stuff is in 2,700 lines.
Boilerplate and abstraction indirections
and all these things are just bad code.
Well, let's talk about good code and bad code.
There's a, I would say, I don't know,
for generic scripts that I write just off-hand,
like 80% of it is written by GPT.
Just like quick off-hand stuff.
So not like libraries, not like performing code,
not stuff for robotics, and so on, just quick stuff.
Because your basics, so much of programming
is doing some, yeah, boilerplate,
but to do so efficiently and quickly,
because you can't really automate it fully
with a generic kind of ID type of recommendation,
something like this, you do need to have

some of the complexity of language models.
Yeah, I guess if I was really writing,
maybe today, if I wrote a lot of data parsing stuff,
I mean, I don't play CTFs anymore,
but if I still play CTFs, a lot of it is just like,
you have to write a parser for this data format.
I wonder, or like admin of code,
I wonder when the models are gonna start to help
with that kind of code, and they may.
They may, and the models also may help you with speed.
Yeah.
The models are very fast, but where the models won't,
my programming speed is not at all limited
by my typing speed.
And in very few cases, it is.
Yes, if I'm writing some script
to just like parse some weird data format,
sure, my programming speed is limited by my typing speed.
What about looking stuff up?
Because that's essentially a more efficient lookup, right?
You know, when I was at Twitter,
I tried to use ChatGPT to like ask some questions,
like, what's the API for this?
And it would just hallucinate.
It would just give me completely made up API functions
that sounded real.
What, do you think that's just a temporary kind of stage?
No.
You don't think it'll get better and better
and better and this kind of stuff,
because like it only hallucinates stuff in the edge cases.
Yes.
If you write generic code, it's actually pretty good.
Yes, if you are writing an absolute basic,
like React app with a button,
it's not gonna hallucinate, sure.
No, there's kind of ways to fix the hallucination problem.
I think Facebook has an interesting paper.
It's called Atlas.
And it's actually weird the way that we do language models
right now where all of the information is in the weights.
And human brains don't really like this.
It's like a hippocampus and a memory system.

So why don't LLMs have a memory system?
And there's people working on them.
I think future LLMs are gonna be like smaller,
but are going to run looping on themselves
and are going to have retrieval systems.
And the thing about using a retrieval system
is you can cite sources explicitly.
Which is really helpful to integrate the human
into the loop of the thing,
because you can go check the sources
and you can investigate it.
So whenever the thing is hallucinating,
you can like have the human supervision.
That's pushing it towards level two kind of journey.
That's gonna kill Google.
Wait, which part?
When someone makes an LLM that's capable
of citing its sources, it will kill Google.
LLM that's citing its sources
because that's basically a search engine.
That's what people want in a search engine.
But also Google might be the people that build it.
Maybe.
I'll put ads on them.
I'd count them out.
Why is that?
What do you think?
Who wins this race?
We got, who are the competitors?
All right.
We got TinyCorp.
I don't know if that's,
yeah, I mean you're a legitimate competitor in that.
I'm not trying to compete on that.
You're not.
No, not as a competitor.
This can accidentally stumble into that competition.
Maybe.
You don't think you might build a search engine
to replace Google search?
When I started comma, I said over and over again,
I'm going to win self-driving cars.
I still believe that.

I have never said I'm going to win search
with the TinyCorp and I'm never going to say that
because I won't.
The night is still young.
We don't, you don't know how hard is it to win search
in this new route.
It feels, I mean, one of the things that ChadGPT
kind of shows that there could be a few interesting tricks
that really have, that create a really compelling product.
Some startup's going to figure it out.
I think, I think if you ask me,
like Google's still the number one webpage,
I think by the end of the decade,
Google won't be the number one webpage anymore.
So you don't think Google,
because of the, how big the corporation is?
Look, I would put a lot more money on Mark Zuckerberg.
Why is that?
Because Mark Zuckerberg's alive.
Like this is the old Paul Graham essay.
Startups are either alive or dead.
Google's dead.
Facebook is alive.
Versus Facebook is alive, that's alive.
Meta. Meta.
You see what I mean?
Like that's just like, like, like Mark Zuckerberg.
This is Mark Zuckerberg reading that Paul Graham essay
and being like, I'm going to show everyone how alive we are.
I'm going to change the name.
So you don't think there's this gutsy pivoting engine
that like Google doesn't have that,
the kind of engine that a startup has like constantly.
You know what?
Being alive, I guess.
When I listened to your Sam Altman podcast,
he talked about the button.
Everyone who talks about AI talks about the button,
the button to turn it off, right?
Do we have a button to turn off Google?
Is anybody in the world capable of shutting Google down?
What does that mean exactly?
The company or the search engine?

So we shut the search engine down.
Could we shut the company down?
Either.
Can you elaborate on the value of that question?
Does Sundar Pashai have the authority
to turn off Google.com tomorrow?
Who has the authority?
That's a good question, right?
Does anyone?
Does anyone?
Yeah, I'm sure.
Are you sure?
No, they have the technical power,
but do they have the authority?
Let's say Sundar Pashai made this his sole mission.
Came into Google tomorrow and said,
I'm going to shut Google.com down.
Yeah.
I don't think he'd keep his position too long.
And what is the mechanism
by which he wouldn't keep his position?
Well, the boards and shares and corporate undermining
and oh my God, our revenue is zero now.
Okay.
So what I mean, what's the case you're making here?
So the capitalist machine prevents you
from having the button.
Yeah.
And it will have it.
I mean, this is true for the AIs too, right?
There's no turning the AIs off.
There's no button.
You can't press it.
Now, does Mark Zuckerberg have that button
for Facebook.com?
Yeah, it's probably more.
I think he does.
I think he does.
This is exactly what I mean
and why I bet on him so much more than I bet on Google.
I guess you could say Elon has some other stuff.
Oh, Elon has the button.
Yeah.

Elon, does Elon, can Elon fire the missiles?
Can he fire the missiles?
I think some questions that better are unasked.
Right?
I mean, you know, a rocket and an ICBM,
oh, you're a rocket that can land anywhere.
Is that an ICBM?
Well, yeah, you know, don't ask too many questions.
My God.
But the positive side of the button
is that you can innovate aggressively is what you're saying,
which is what's required with training LLM
into a search engine.
I would bet on a startup.
Because it's so easy, right?
I'd bet on something that looks like mid-journey,
but for search.
Just is able to set source of the loop on itself.
I mean, it just feels like one model can take off, right?
And that nice wrapper and some of it, I mean,
it's hard to like create a product
that just works really nicely, stably.
The other thing that's gonna be cool
is there is some aspect of a winner take all effect, right?
Like once someone starts deploying a product
that gets a lot of usage,
and you see this with OpenAI,
they are going to get the data set
to train future versions of the model.
Yeah.
They are going to be able to, right?
You know, I was actually at Google Image Search
when I worked there like almost 15 years ago now.
How does Google know which image is an apple?
And I said the metadata and they're like,
yeah, that works about half the time.
How does Google know?
You'll see they're all apples on the front page
when you search Apple.
And I don't know, I didn't come up with the answer.
The guys are like, well, it's what people click on
when they search Apple.
I'm like, oh yeah.

Yeah, yeah, that data is really, really powerful.
It's the human supervision.
What do you think are the chances?
What do you think in general that Llama was open sourced?
I just did a conversation with Mark Zuckerberg
and he's all in on open source.
Who would have thought that Mark Zuckerberg
would be the good guy?
I mean it.
Who would have thought anything in this world?
It's hard to know.
But open source to you ultimately is a good thing here.
Undoubtedly.
You know, what's ironic about all these AI safety people
is they are going to build the exact thing they fear.
These, we need to have one model that we control and align.
This is the only way you end up paper clipped.
There's no way you end up paper clipped
if everybody has an AI.
So open sourcing is the way to fight the paper clip,
Maximizer.
Absolutely.
It's the only way.
You think you're going to control it?
You're not going to control it.
So the criticism you have for the AI safety folks
is that there is a belief and a desire for control.
Yeah.
And that belief and desire for centralized control
of dangerous AI systems is not good.
Sam Altman won't tell you that GPT-4
has 220 billion parameters
and is a 16-way mixture model with eight sets of weights.
Who did you have to murder to get that information?
All right.
I mean, look, everyone at OpenAI knows
what I just said was true, right?
Now, ask the question really, you know,
it upsets me when I, like GPT-2,
when OpenAI came out with GPT-2
and raised a whole fake AI safety thing about that,
I mean, now the model is laughable.
Like they used AI safety to hype up their company

and it's disgusting.
Or the flip side of that is they used
a relatively weak model in retrospect
to explore how do we do AI safety correctly?
How do we release things?
How do we go through the process?
I don't know if...
Sure, all right, all right, all right, all right.
That's the...
I don't know how much hype there is.
That's the charitable interpretation.
I don't know how much hype there is in AI safety, honestly.
Oh, there's so much.
At least not to play there.
I don't know.
Maybe Twitter's not real.
Come on, in terms of hype.
I mean, I don't...
I think OpenAI has been finding an interesting balance
between transparency and putting value on AI safety.
You don't think...
You think just go a lot open source.
So do with Lama.
Absolutely, yeah.
So do like open source, this is a tough question,
which is open source both the base,
the foundation model and the fine-tuned one.
So the model that can be ultra-racist and dangerous
and tell you how to build a nuclear weapon...
Oh my God, have you met humans, right?
Like half of these AI...
I haven't met most humans.
This allows you to meet every human.
Yeah, I know, but half of these AI alignment problems
are just human alignment problems.
And that's what's also so scary about the language they use.
It's like, it's not the machines you want to align, it's me.
But here's the thing, it makes it very accessible
to ask very questions where the answers have dangerous consequences
if you were to act on them.
I mean, yeah, welcome to the world.
Well, no, for me, there's a lot of friction.
If I want to find out how to, I don't know, blow up something.

No, there's not a lot of friction, it's so easy.
No, like what do I search?
Do I use Bing or do I use...
No, there's like lots of stuff.
No, it feels like I have to keep clicking a lot of this.
Anyone who's stupid enough to search for how to blow up a building
in my neighborhood is not smart enough to build a bomb, right?
Are you sure about that?
Yes.
I feel like a language model makes it more accessible
for that person who's not smart enough to do...
They're not going to build a bomb, trust me.
The people who are incapable of figuring out how to ask that question
a bit more academically and get a real answer from it
are not capable of procuring the materials,
which are somewhat controlled to build a bomb.
No, I think LLM makes it more accessible to people with money
without the technical know-how, right?
To build a...
Like, do you really need to know how to build a bomb to build a bomb?
You can hire people, you can file a...
Or you can hire people to build a...
You know what, I was asking this question on my stream,
like, can Jeff Bezos hire a hitman?
Probably not.
But a language model can probably help you out?
Yeah, and you'll still go to jail, right?
Like, it's not like the language model is God.
Like, the language model...
It's like, you literally just hired someone on Fiverr.
But you...
But, okay, okay, GPT-4, in terms of finding a hitman,
it's like asking Fiverr how to find a hitman.
I understand.
But don't you think...
As in WikiHow, you know?
WikiHow.
But don't you think GPT-5 will be better?
Because don't you think that information is out there on the internet?
I mean, yeah, and I think that if someone is actually serious
enough to hire a hitman or build a bomb,
they'd also be serious enough to find the information.
I don't think so.

I think it makes it more accessible.
If you have enough money to buy a hitman,
I think it decreases the friction of how hard is it to find that kind of hitman.
I honestly think there's a jump in ease and scale of how much harm you can do.
And I don't mean harm with language.
I mean, harm with actual violence.
What you're basically saying is like,
okay, what's going to happen is these people who are not intelligent.
Are going to use machines to augment their intelligence.
And now, intelligent people and machines, intelligence is scary.
Intelligent agents are scary.
When I'm in the woods, the scariest animal to meet is a human, right?
No, no, no.
There's look, there's like nice California humans.
Like I see you're wearing like, you know, street clothes and Nikes.
All right, fine.
You look like you've been a human who's been in the woods for a while.
Yeah.
I'm more scared of you than a bear.
That's what they say about the Amazon.
When you go to the Amazon, it's the human tribes.
Oh, yeah.
So intelligence is scary, right?
So to ask this question in a generic way, you're like,
what if we took everybody who, you know, maybe has ill intention,
but is not so intelligent and gave them intelligence, right?
So we should have intelligence control.
Of course, we should only give intelligence to good people.
And that is the absolutely horrifying idea.
So to you, the best defense is actually,
the best defense is to give more intelligence
to the good guys and intelligence.
Give intelligence to everybody.
Give intelligence to everybody.
You know what?
And it's not even like guns, right?
Like people say this about guns.
You know, what's the best defense against a bad guy with a gun?
Good guy with a gun.
Like I kind of subscribe to that,
but I really subscribe to that with intelligence.
Yeah.
And to find them out the way I agree with you,

but there's just feels like so much uncertainty
and so much can happen rapidly that you can lose a lot of control
and you can do a lot of damage.
Oh, no, we can lose control.
Yes.
Thank God.
Yeah.
I hope we can, I hope they lose control.
I'd want them to lose control more than anything else.
I think when you lose control, you can do a lot of damage,
but you can do more damage when you centralize
and hold on to control is the point.
Centralized and held control is tyranny.
Right?
I will always, I don't like anarchy either,
but I'll always take anarchy over tyranny.
Anarchy, you have a chance.
This human civilization we've got going on is quite interesting.
I mean, I agree with you.
So to you, open source is the way forward here.
Is the way forward here.
So you admire what Facebook is doing here
or what Metta is doing with the release of them.
A lot.
Yeah.
A lot.
I lost $80,000 last year investing in Metta
and when they released Lama, I'm like, yeah, whatever, man.
That was worth it.
It was worth it.
Do you think Google and OpenAI or Microsoft will match
what Metta is doing or not?
So if I were a researcher, why would you want to work at OpenAI?
Like, you know, you're just, you're on the bad team.
Like, I mean it.
Like, you're on the bad team who can't even say
that GPT-4 has 220 billion parameters.
So close source to use the bad team.
Not only close source.
I'm not saying you need to make your model weights open.
I'm not saying that.
I totally understand we're keeping our model weights closed
because that's our product, right?

That's fine.
I'm saying like, because of AI safety reasons,
we can't tell you the number of billions of parameters in the model.
That's just the bad guys.
Just because you're mocking AI safety doesn't mean it's not real.
Oh, of course.
Is it possible that these things can really do a lot of damage
that we don't know about?
Oh my God, yes.
Intelligence is so dangerous.
Be it human intelligence or machine intelligence.
Intelligence is dangerous.
But machine intelligence is so much easier to deploy at scale,
like rapidly.
Like what, okay.
If you have human-like bots on Twitter.
All right.
And you have like a thousand of them create a whole narrative.
Like you can manipulate millions of people.
But you mean like the intelligence agencies in America are doing right now?
Yeah, but they're not doing it that well.
It feels like you can do a lot.
They're doing it pretty well.
What?
I think they're doing a pretty good job.
I suspect they're not nearly as good as a bunch of GPT fuel bots could be.
Well, I mean, of course they're looking into the latest technologies
for control of people, of course.
But I think there's a George Haas type character that can do a better job
than the entirety of them.
You don't think so?
No way.
No, and I'll tell you why the George Haas character can't.
And I thought about this a lot with hacking, right?
Like I can find exploits in web browsers.
I probably still can.
I mean, I was better when I was 24, but the thing that I lack
is the ability to slowly and steadily deploy them over five years.
And this is what intelligence agencies are very good at, right?
Intelligence agencies don't have the most sophisticated technology.
They just have.
Endurance?
Endurance.

And yeah, the financial backing and the infrastructure for the endurance.
So the more we can decentralize power, like you could make an argument, by the way,
that nobody should have these things.
And I would defend that argument.
I would, like you're saying, look, LLMs and AI and machine intelligence
can cause a lot of harm, so nobody should have it.
And I will respect someone philosophically with that position,
just like I will respect someone philosophically with the position
that nobody should have guns, right?
But I will not respect philosophically with only the trusted authorities
should have access to this.
Who are the trusted authorities?
You know what?
I'm not worried about alignment between AI company and their machines.
I'm worried about alignment between me and AI company.
What do you think Eliezer Yatkowski would say to you?
Because he's really against open source.
I know.
And I thought about this.
I thought about this.
And I think this comes down to a repeated misunderstanding
of political power by the rationalists.
Interesting.
I think that Eliezer Yatkowski is scared of these things.
And I am scared of these things too.
Everyone should be scared of these things.
These things are scary.
But now you ask about the two possible futures.
One where a small trusted centralized group of people has them.
And the other where everyone has them.
And I am much less scared of the second future than the first.
Well, there's a small trusted group of people
that have control of our nuclear weapons.
There's a difference.
Again, a nuclear weapon cannot be deployed tactically.
And a nuclear weapon is not a defense against a nuclear weapon.
Except maybe in some philosophical mind game kind of way.
But AI is different in different how exactly.
Okay.
Let's say the intelligence agency deploys a million bots on Twitter
or a thousand bots on Twitter to try to convince me of a point.
Imagine I had a powerful AI running on my computer saying,
okay, nice Psyop.

Nice Psyop.
Nice Psyop.
Okay.
Here's a Psyop.
I filtered it out for you.
Yeah.
I mean, so you have fundamental hope for that, for the defense of Psyop.
I'm not even like, I don't even mean these things in like truly horrible ways.
I mean these things in straight up like ad blocker.
Right?
Yeah.
Straight up ad blocker.
Right?
I don't want ads.
Yeah.
But they are always finding, you know, imagine I had an AI
that could just block all the ads for me.
So you believe in the power of the people to always create an ad blocker?
Yeah.
I mean, I kind of share that belief.
I have, that's one of the deepest optimisms I have is just like,
there's a lot of good guys.
So to give, you shouldn't hand pick them, just throw out powerful technology out there
and the good guys will outnumber and outpower the bad guys.
Yeah.
I'm not even going to say there's a lot of good guys.
I'm saying that good outnumber is bad.
Right?
Good outnumber is bad.
In skill and performance.
Yeah.
Definitely in skill and performance, probably just a number too.
Probably just in general.
I mean, if you believe philosophically in democracy, you obviously believe that.
That good outnumber is bad.
And like the only, if you give it to a small number of people,
there's a chance you gave it to good people, but there's also a chance you gave it to bad people.
If you give it to everybody, well, if good outnumber is bad,
then you definitely gave it to more good people than bad.
That's really interesting.
So that's on the safety grounds, but then also, of course, there's other motivations like,
you don't want to give away your secret sauce.
Well, that's, I mean, I look, I respect capitalism.

I don't think that.
I think that it would be polite for you to make model architectures open source and fundamental breakthroughs open source.
I don't think you have to make a way to open source.
You know, what's interesting is that like, there's so many possible trajectories in human history where you could have the next Google be open source.
So for example, I don't know if that connection is accurate,
but you know, Wikipedia made a lot of interesting decisions, not to put ads.
Like Wikipedia is basically open source.
You could think of it that way.
Yeah.
And like, that's one of the main websites on the internet.
And like, it didn't have to be that way.
It could have been like Google could have created Wikipedia, put ads on it.
You could probably run amazing ads now on Wikipedia.
You wouldn't have to keep asking for money, but it's interesting, right?
So Lama, open source Lama, derivatives of open source Lama might win the internet.
I sure hope so.
I hope to see another era.
You know, the kids today don't know how good the internet used to be.
And I don't think this is just, come on, like everyone's nostalgic for their past,
but I actually think the internet before small groups of weaponized corporate and government interests took it over, was a beautiful place.
You know, those small number of companies have created some sexy products,
but you're saying overall, in the long arc of history, the centralization of power they have like suffocated the human spirit at scale.
Here's a question to ask about those beautiful sexy products.
Imagine 2000 Google to 2010 Google, right?
A lot changed.
We got maps.
We got Gmail.
We lost a lot of products too, I think.
From, yeah, I mean, somewhere probably.
We've got Chrome, right?
And now let's go from 2010, we got Android.
Now let's go from 2010 to 2020.
What does Google have?
Well, search engine, maps, mail, Android and Chrome.
Oh, I see.
Yeah.
The internet was this.
You know, I was Times First of the Year in 2006.
Yeah.

I love this.
Yeah, it's you was Times First of the Year in 2006, right?
Like, that's, you know, so quickly did people forget.
And I think some of it's social media.
I think some of it, I hope, look, I hope that I don't, it's possible that some very sinister things happen.
I don't know.
I think it might just be like the effects of social media.
But something happened in the last 20 years.
Oh, okay.
So you're just being an old man who's worried about the, I think there's always, it goes, it's the cycle thing.
It's ups and downs.
And I think people rediscover the power of distributed of decentralized.
Yeah.
I mean, that's kind of like what the whole cryptocurrency is trying, like that that,
I think crypto is just carrying the flame of that spirit of like, stuff should be decentralized.
It's just such a shame that they all got rich.
You know?
Yeah.
If you took all the money out of crypto, it would have been a beautiful place.
Yeah.
But no, I mean, these people, you know, they, they, they sucked all the value out of it and took it.
Yeah.
Money kind of corrupts the mind somehow.
It becomes a drug.
You corrupted all of crypto.
You had coins worth billions of dollars that had zero use.
You still have hope for crypto?
Sure.
I have hope for the ideas.
I really do.
Yeah.
I mean, you know,
I want the US dollar to collapse.
I do.
George Haughts.
Well, let me sort of on, on the ASAT, do you think there's some interesting questions there, though, to solve for the open source community in this case?
So like alignment, for example, or the control problem, like if you really have super powerful, you said it's scary.
Oh yeah.
What do we do with it?

So not, not control, not centralized control, but like,
if you were, then you're going to see some guy or gal release a super powerful language model, open source.
And here you are, George Haughts, thinking, holy shit.
Okay.
What ideas do I have to combat this thing?
So what ideas would you have?
I am so much not worried about the machine independently doing harm.
That's what some of these AI safety people seem to think.
They somehow seem to think that the machine like independently is going to rebel against its creator.
So you don't think he'll find autonomy?
No, this is sci-fi B movie garbage.
Okay.
What if the thing writes code, basically writes viruses?
If the thing writes viruses, it's because the human told it to write viruses.
Yeah, but there's some things you can't like put back in the box.
That's the whole, that's kind of the whole point.
Is it kind of spreads, give it access to the internet, it spreads, installs itself.
Modifies your shit.
B, B, B, B plot sci-fi, not real.
Unless I'm trying to work, I'm trying to get better at my plot writing.
The thing that worries me, I mean, we have a real danger to discuss and that is bad humans using the thing to do whatever bad unaligned AI thing you want.
But this goes to your previous concern that who gets to define who's a good human, who's a bad human.
Nobody does, we give it to everybody.
And if you do anything besides give it to everybody, trust me, the bad humans will get it.
And that's who gets power.
It's always the bad humans who get power.
Okay, power.
And power turns even slightly good humans to bad.
Sure.
That's the intuition you have.
I don't know.
I don't think everyone, I don't think everyone.
I just think that like, here's the saying that I put in one of my blog posts.
When I was in the hacking world, I found 95% of people to be good and 5% of people to be bad.
Like just who I personally judged as good people and bad people.
They believed about good things for the world.
They wanted like flourishing and they wanted growth and they wanted things I consider good.
Right?
I came into the business world with comma and I found the exact opposite.

I found 5% of people good and 95% of people bad.
I found a world that promotes psychopathy.
I wonder what that means.
I wonder if that's anecdotal or if there's truth to that, there's something about capitalism at the core that promotes the people that run capitalism, that promotes psychopathy.
That saying may of course be my own biases.
Right?
That may be my own biases that these people are a lot more aligned with me than these other people.
Right?
Yeah.
So, I can certainly recognize that, but in general, I mean, this is like the common sense maximum, which is the people who end up getting power are never the ones you want with it.
But do you have a concern of super intelligent AGI, open sourced, and then what do you do with that?
I'm not saying control it.
It's open source.
What do we do with this human species?
If that's not up to me, I mean, you know, like I'm not a central planner.
No, not central planner, but you'll probably tweet there's a few days left to live for the human species.
I have my ideas of what to do with it and everyone else has their ideas of what to do with it.
And may the best ideas win.
But at this point, do you brainstorm?
Because it's not regulation.
It could be decentralized regulation where people agree that this is just like we create tools that make it more difficult for you to maybe make it more difficult for code to spread, you know, antivirus software, this kind of thing.
No, you're saying that you should build AI firewalls?
That sounds good.
You should definitely be running an AI firewall.
Yeah, right.
You should be running an AI firewall to your mind.
Right.
You're constantly under, you know.
Such an interesting idea.
It's like info wars, man.
Like, I don't know if you're being sarcastic or not.
No, I'm dead serious.
But I think there's power to that.
It's like, how do I protect my mind from influence of human like or super human intelligent bots?
I would pay so much money for that product.
I would pay so much money for that product.

I would, you know, how much money I'd pay just for a spam filter that works?
Well, on Twitter, sometimes I would like to have a protection mechanism for my mind
from the outrage mobs because they feel like bot like behavior.
It's like, there's a large number of people that will just grab a viral narrative
and attack anyone else that believes otherwise.
And it's like, whenever someone's telling me some story from the news, I'm always like,
I don't want to hear it.
It's CIA app, bro.
It's a CIA app, bro.
Like, it doesn't matter if that's true or not.
It's just trying to influence your mind.
You're repeating an ad to me with the viral mobs of this.
Is it like, yeah.
To me, a defense against those mobs is just getting multiple perspectives
always from sources that make you feel kind of like you're getting smarter.
And just actually just basically feels good.
Like a good documentary just feels, something feels good about it.
It's well done.
It's like, oh, okay.
I never thought of it this way.
This just feels good.
Sometimes the outrage mobs, even if they have a good point behind it,
when they're mocking and derisive and just aggressive,
you're with us or against us, this fucking.
This is why I delete my tweets.
Yeah.
Why'd you do that?
I was, I miss your tweets.
You know what it is?
The algorithm promotes toxicity.
And like, I think Elon has a much better chance of fixing it than the previous regime.
Yeah.
But to solve this problem, to solve, like to build a social network that is actually not toxic
without moderation.
Like not to stick but care it.
So like where people look for goodness,
so make it catalyze the process of connecting cool people and being cool to each other.
Yeah.
Without ever censoring.
Without ever censoring.
And like Scott Alexander has a blog post I like where he talks about moderation is not censorship.
Like all moderation you want to put on Twitter,
you could totally make this moderation like just a,

you don't have to block it for everybody.
You can just have like a filter button that people can turn off if they would like say search for Twitter.
Like someone could just turn that off.
So like, but then you'd like take this idea to an extreme.
Well, the network should just show you, this is a couch surfing CEO thing.
If it shows you right now, these algorithms are designed to maximize engagement.
Well, it turns out outrage maximizes engagement.
Quirk of human, quirk of the human mind.
Right.
Just this, I fall for it, everyone falls for it.
So yeah, you got to figure out how to maximize for something other than engagement.
And I actually believe that you can make money with that too.
So it's not, I don't think engagement is the only way to make money.
I actually think it's incredible that we're starting to see, I think again,
Elon's doing so much stuff right with Twitter like charging people money.
As soon as you charge people money, they're no longer the product.
They're the customer.
And then they can start building something that's good for the customer and not good for the other customer, which is the ad agencies.
As in, as in picked up steam.
I pay for Twitter.
It doesn't even get me anything.
It's my donation to this new business model, hopefully working out.
Sure.
But you know, you, for this business model to work, it's like most people should be signed up to Twitter.
And so the way it was, there was something perhaps not compelling or something like this to people.
I think you need most people at all.
I think that, why do I need most people, right?
Don't make an 8,000 person company, make a 50 person company.
Well, so speaking of which, you worked at Twitter for a bit.
I did.
As an intern.
The world's greatest intern.
All right.
There's been better.
There's been better.
Tell me about your time at Twitter.
How did it come about?
And what did you, did you learn from the experience?
So I deleted my first Twitter in 2010.

I had over 100,000 followers back when that actually meant something.
And I just saw, you know, my co-worker summarized it well.
He's like, whenever I see someone's Twitter page, I either think the same of them or less of them.
I never think more of them, right?
Like, you know, I don't want to mention any names, but like some people who like, you know, maybe you would like read their books and you would respect them.
You see them on Twitter and you're like, okay, dude.
Yeah.
But there are some people with same.
You know who I respect a lot are people that just post really good technical stuff.
Yeah.
And I guess, I don't know, I think I respect them more for it because you realize, oh, this wasn't, there's like so much depth to this person, to their technical understanding of so many different topics.
Okay.
So I try to follow people.
I try to consume stuff that's technical machine learning content.
There's probably a few of those people.
And the problem is inherently what the algorithm rewards, right?
And people think about these algorithms.
People think that they are terrible, awful things.
And, you know, I love that Elon open sourced it because, I mean, what it does is actually pretty obvious.
It just predicts what you are likely to retweet and like and linger on.
That's what all these algorithms do.
So we take talk to us.
So all these recommendation I just do.
And it turns out that the thing that you are most likely to interact with is outrage.
And that's a quirk of the human condition.
I mean, and there's different flavors of outrage.
It doesn't have to be, it could be mockery.
You'd be outraged.
The topic of outrage could be different.
It could be an idea.
It could be a person.
It could be, and maybe there's a better word than outrage.
It could be drama.
Sure.
Drama.
That's kind of stuff.
Yeah.
But it doesn't feel like when you consume it, it's a constructive thing for the individuals that consume it in the long term.

Yeah.
So my time there, I absolutely couldn't believe, you know, I got crazy amount of hate, you know, just on Twitter for working at Twitter.
It seemed like people associated with this.
I think maybe you were exposed to some of this.
So connection to Elon or is it working at Twitter?
Twitter and Elon, like the whole.
Elon's gotten a bit spicy during that time.
A bit political, a bit.
Yeah.
Yeah.
You know, I remember one of my tweets, it was never go full of Republican and Elon liked it.
Oh boy.
Yeah.
I mean, there's a roller coaster of that, but being political on Twitter.
Yeah.
Boy.
Yeah.
And also being just attacking anybody on Twitter, it comes back at you harder.
And if his political ant attacks.
Sure.
Sure, absolutely.
And then letting sort of de-platform people back on even adds more fun to the beautiful chaos.
I was hoping and like I remember when Elon talked about buying Twitter like six months earlier, he was talking about like a principled commitment to free speech.
And I'm a big believer and fan of that.
I would love to see an actual principled commitment to free speech.
Of course, this isn't quite what happened.
Instead of the oligarchy deciding what to ban, you had a monarchy deciding what to ban, right?
Instead of, you know, all the Twitter files, shadow, really, the oligarchy just decides what?
Cloth masks are ineffective against COVID.
That's a true statement.
Every doctor in 2019 knew it.
And now I'm banned on Twitter for saying it.
Interesting.
Oligarchy.
So now you have a monarchy and, you know, he bans things he doesn't like.
So, you know, it's just different.
It's different power and like, you know, maybe I align more with him than with the oligarchy.
But it's not free speech absolutism.
But I feel like being a free speech absolutist on a social network requires you to also have tools for the individuals to control what they consume easier.
Like, not censor, but just like control, like, oh, I'd like to see more cats and less politics.

And this isn't even remotely controversial.
This is just saying you want to give paying customers for a product what they want.
Yeah.
And not through the process of censorship, but through the process of like...
Well, it's individualized, right?
It's individualized, transparent censorship, which is honestly what I want.
What is an ad block or it's individualized, transparent censorship, right?
Yeah, but censorship is a strong word and people are very sensitive, too.
I know, but, you know, I just use words to describe what they functionally are.
And what is an ad block or it's just censorship.
Well, when I look at you right now...
But I love what you're censoring.
I'm looking at you.
I'm censoring everything else out when my mind is focused on you.
That's, you can use the word censorship that way, but usually when people get very sensitive about the censorship thing, I think when you have, when anyone is allowed to say anything, you should probably have tools that maximize the quality of the experience for individuals.
It's like, you know, for me, like what I really value, boy, would be amazing to somehow figure out how to do that.
I love disagreement and debate and people who disagree with each other disagree with me, especially in the space of ideas, but the high quality ones.
So not derision, right?
Maslow's hierarchy of argument.
I think that's a real word for it.
Probably.
Yeah.
There's just a way of talking that's like snarky and so on that somehow is, gets people on Twitter and they get excited and so on.
You have like ad hominem refuting the central point.
I've like seen this as an actual pyramid somewhere.
Yeah.
It's, yeah.
And it's like all of it, all the wrong stuff is attractive to people.
I mean, we can just try to classify it to absolutely say what level of Maslow's hierarchy of argument are you at.
Yeah.
And if it's ad hominem, like, okay, cool.
I turned on the no ad hominem filter.
I wonder if there's a social network that will allow you to have that kind of filter.
Yeah.
So here's a problem with that.
It's not going to win in a free market.
Yeah.

What wins in a free market is all television today is reality television because it's engaging, right?
If engaging is what wins in a free market, right?
So it becomes hard to keep these other more nuanced values.
Well, okay.
So that's the experience of being on Twitter, but then you got a chance to also
together with other engineers and with Elon sort of look,
brainstorm when you step into a code base has been around for a long time.
You know, there's other social networks, you know, Facebook, this is old code bases
and you step in and see, okay, how do we make with a fresh mind progress on this code base?
Like what did you learn about software engineering, about programming from just experiencing that?
So my technical recommendation to Elon and I said this on the Twitter space is afterward.
I said this many times during my brief internship.
Was that you need refactors before features.
This code base was, and look, I've worked at Google, I've worked at Facebook.
Facebook has the best code, then Google, then Twitter.
And you know what?
You can know this because look at the machine learning frameworks, right?
Facebook released PyTorch, Google released TensorFlow and Twitter released.
Okay.
So, you know, it's a proxy, but yeah, the Google code base is quite interesting.
There's a lot of really good software engineers there, but the code base is very large.
The code base was good in 20 and 2005, right?
It looks like 2005.
I've got so many products, so many teams, right?
It's very difficult to, I feel like Twitter does less, like obviously much less than Google
in terms of like the set of features, right?
So like it's, I can imagine the number of software engineers that could recreate
Twitter is much smaller than to recreate Google.
Yeah.
I still believe in the amount of hate I got for saying this,
that 50 people could build and maintain Twitter pretty.
What's the nature of the hate?
Comfortably.
But you don't know what you're talking about?
You know what it is?
And it's the same, this is my summary of like the hate I get on Hacker News.
It's like, when I say I'm going to do something, they have to believe that it's impossible.
Because if doing things was possible, they'd have to do some soul searching and ask the question, why didn't they do anything?
So when you say, when you say, well, there's a core truth to that.
Yeah.

So when you say, I'm going to solve self-driving, people go like, what are your credentials?
What the hell are you talking about?
What is, this is an extremely difficult problem.
Of course, you're a noob that doesn't understand the problem deeply.
I mean, that that was the same nature of hate that probably Elon got when he first talked about autonomous driving.
But there's pros and cons to that because there is experts in this world.
No, but the mockers aren't experts.
The people who are mocking are not experts with carefully reasoned arguments about why you need 8,000 people to run a bird app, but the people are going to lose their jobs.
Well, that, but also there's the software engineers that probably criticize, no, it's a lot more complicated than you realize, but maybe it doesn't need to be so complicated.
Some people in the world like to create complexity.
Some people in the world thrive under complexity like lawyers, right?
Lawyers want the world to be more complex because you need more lawyers, you need more legal hours.
I think that's another.
If there's two great evils in the world, it's centralization and complexity.
Yeah. And the one of the sort of hidden side effects of software engineering is like finding pleasure and complexity.
I mean, I don't remember just taking all the software engineering courses and just doing programming and just coming up in this object-oriented programming kind of idea.
Not often do people tell you like, do the simplest possible thing.
Like a professor, a teacher is not going to get in front and like, this is the simplest way to do it.
They'll say like, this is the right way and the right way, at least for a long time, you know, especially I came up with like Java, right?
There's so much boilerplate, so much like so many classes, so many like designs and architectures and so on, like planning for features far into the future and planning poorly and all this kind of stuff. And then there's this like code base that follows you along and puts pressure on you and nobody knows what like parts, different parts do, which slows everything down.
There's a kind of bureaucracy that's instilled in the code as a result of that.
But then you feel like, oh, well, I follow good software engineering practices.
It's an interesting trade-off because then you look at like the get-on-ness of like Pearl and the old like, how quickly you could just write a couple lines and just get stuff done.
That trade-off is interesting or bash or whatever, these kind of ghetto things you can do in Linux.
One of my favorite things to look at today is how much do you trust your tests, right?
We've put a ton of effort in comma and I put a ton of effort in tiny grad into making sure if you change the code and the tests pass that you didn't break the code.
Now, this obviously is not always true, but the closer that is to true, the more you trust your tests, the more you're like, oh, I got a pull request and the tests pass, I feel okay to merge that, the faster you can make progress.
So you're always programming with tests in mind, developing tests with that in mind,

that if it passes, it should be good.
And Twitter had a...
Not that.
So...
It was impossible to make progress in the code base.
What other stuff can you say about the code base that made it difficult?
What are some interesting sort of quirks?
Broadly speaking from that, compared to just your experience with comma and everywhere else.
The real thing that I spoke to a bunch of individual contributors at Twitter,
and I just stashed, I'm like, okay, so what's wrong with this place?
Why does this code look like this?
And they explained to me what Twitter's promotion system was.
The way that you got promoted to Twitter was you wrote a library that a lot of people used.
Right?
So some guy wrote an nginx replacement for Twitter.
Why does Twitter need an nginx replacement?
What was wrong with nginx?
Well, you see, you're not going to get promoted if you use nginx.
But if you write a replacement and lots of people start using it as the Twitter front end
for their product, then you're going to get promoted, right?
So interesting because from an individual perspective, how do you incentivize...
How do you create the kind of incentives that will lead to a great code base?
Okay, what's the answer to that?
So what I do at Comma and at TinyCorp is you have to explain it to me.
You have to explain to me what this code does, right?
And if I can sit there and come up with a simpler way to do it, you have to rewrite it.
You have to agree with me about the simpler way.
Obviously, we can have a conversation about this.
It's not a dictatorial, but if you're like, wow, wait, that actually is way simpler.
Like the simplicity is important, right?
But that requires people that overlook the code at the highest levels to be like, okay.
It requires technical leadership, you trust.
Yeah, technical leadership.
So managers or whatever should have technical savvy, deep technical savvy.
Managers should be better programmers than the people who they manage.
And that's not always obvious, trivial to create, especially large companies.
Managers get soft.
And this is just, I've instilled this culture at Comma, and Comma has better programmers than
me who work there.
But again, I'm like the old guy from Goodwill Hunting.
It's like, look, man, I might not be as good as you, but I can see the difference between
me and you, right?
And like, this is what you need.

This is what you need at the top.
Or you don't necessarily need the manager to be the absolute best.
I shouldn't say that, but they need to be able to recognize skill.
Yeah, and have good intuition, intuition that's laden with wisdom from all the battles
of trying to reduce complexity and code bases.
I took a political approach at Comma, too, that I think is pretty interesting.
I think Elon takes the same political approach.
Google had no politics.
And what ended up happening is the absolute worst kind of politics took over.
Comma has an extreme amount of politics, and they're all mine, and no dissidence is tolerated.
So it's a dictatorship.
Yep, it's an absolute dictatorship, right?
Elon does the same thing.
Now, the thing about my dictatorship is here are my values.
Yeah, it's just transparent.
It's transparent.
It's a transparent dictatorship, right?
And you can choose to opt in or you get free exit, right?
That's the beauty of companies.
If you don't like the dictatorship, you quit.
So you mentioned rewrite before or refactor before features.
If you were to refactor the Twitter code base, what would that look like?
And maybe also comment on how difficult does it to refactor?
The main thing I would do is, first of all, identify the pieces
and then put tests in between the pieces, right?
So there's all these different Twitter as a microservice architecture.
There's only different microservices.
And the thing that I was working on there, look, George didn't know any JavaScript.
He asked how to fix search, blah, blah, blah, blah, blah.
Look, man, the thing is, I'm upset that the way that this whole thing was portrayed,
because it wasn't like taken by people honestly.
It wasn't like it was taken by people who started out with a bad faith assumption.
And look, I can't like...
And you as a programmer were just being transparent out there, actually having fun.
And this is what programmers should be about.
I love that Elon gave me this opportunity.
Yeah.
Like really, it does.
And he came on the day I quit.
He came on my Twitter spaces afterward and we had a conversation.
I respect that so much.
Yeah.
And it's also inspiring to just engineers and programmers and just...

It's cool.
It should be fun.
The people that were hating on it, it's like, oh, man.
It was fun.
It was fun.
It was stressful.
But I felt like it was a cool point in history.
And I hope I was useful.
I probably kind of wasn't, but maybe I was.
You also were one of the people that kind of made a strong case to refactor.
And that's a really interesting thing to raise.
Maybe that is the right...
The timing of that is really interesting.
If you look at just the development of Autopilot, going from Mobileye to just more...
If you look at the history of Semi-Ton was driving in Tesla, it's more and more...
You could say refactoring or starting from scratch, redeveloping from scratch.
It's refactoring all the way down.
And the question is, can you do that sooner?
Can you maintain product profitability?
What's the right time to do it?
How do you do it?
On any one day, it's like, you don't want to pull off the band-aid.
It's like, everything works.
It's just a little fixed here and there, but maybe starting from scratch.
This is the main philosophy of TinyGrad.
You have never refactored enough.
Your code can get smaller.
Your code can get simpler.
Your ideas can be more elegant.
But would you consider...
Say you were running Twitter development teams, engineering teams,
would you go as far as different programming language?
Just go that far?
The first thing that I would do is build tests.
The first thing I would do is get a CI to where people can trust to make changes.
Before I touched any code, I would actually say no one touches any code.
The first thing we do is we test this code base.
I mean, this is classic.
This is how you approach a legacy code base.
This is how to approach a legacy code base book, we'll tell you.
Then you hope that there's modules that can live on for a while.
And then you add new ones, maybe in a different language or...
Before we add new ones, we replace old ones.

Yeah, meaning replace old ones with something simpler.
We look at this thing that's 100,000 lines and we're like,
well, okay, maybe this did even make sense in 2010,
but now we can replace this with an open source thing.
And we look at this here.
Here's another 50,000 lines.
Well, actually, we can replace this with 300 lines a go.
And you know what?
I trust that the go actually replaces this thing
because all the tests still pass.
So step one is testing.
And then step two is like the programming languages
and afterthought, right?
You know, let a whole lot of people compete,
be like, okay, who wants to rewrite a module,
whatever language you want to write it in,
just the tests have to pass.
And if you figure out how to make the test pass,
but break the site, that's...
We got to go back to step one.
Step one is get tests that you trust
in order to make changes in the code base.
I want to know how hard it is too,
because I'm with you on testing and everything.
From tests to like asserts to code is just covered in this,
because it should be very easy to make rapid changes
and know that's not going to break everything.
And that's the way to do it.
But I wonder how difficult is it to integrate tests
into a code base that doesn't have many of them.
So I'll tell you what my plan was at Twitter.
It's actually similar to something we use at comma.
So at comma, we have this thing called process replay.
And we have a bunch of routes that'll be run through.
So comma is a microservice architecture too.
We have microservices in the driving.
Like we have one for the cameras, one for the sensor,
one for the planner, one for the model.
And we have an API, which the microservices talk to each other with.
We use this custom thing called serial, which uses a ZMQ.
Twitter uses Thrift.
And then it uses this thing called Vanagal,
which is a Scala RPC backend.

But this doesn't really matter.
The Thrift and Vanagal layer was a great place,
I thought, to write tests, to start building something
that looks like process replay.
So Twitter had some stuff that looked kind of like this,
but it wasn't offline.
It was only online.
So you could ship a modified version of it,
and then you could redirect some of the traffic
to your modified version and diff those two.
But it was all online.
Like there was no CI in the traditional sense.
I mean, there was some, but it was not full coverage.
So you can't run all of Twitter offline to test something?
This was another problem.
You can't run all of Twitter.
Period.
Twitter runs in three data centers.
And that's it.
There's no other place you can run Twitter,
which is like, George, you don't understand.
This is modern software development.
No, this is bullshit.
Like, why can't it run on my laptop?
What are you doing?
Twitter can run it.
Yeah.
Okay.
Well, I'm not saying you're going to download the whole database to your laptop,
but I'm saying all the middleware and the front end
should run on my laptop.
That sounds really compelling.
But can that be achieved by a code base that grows over the years?
I mean, the three data centers didn't have to be, right?
Because they're totally different like designs.
The problem is more like, why did the code base have to grow?
What new functionality has been added to compensate for the lines of code that are there?
One of the ways to explain is that the incentive for software developers to move up
in the companies to add code, to add, especially large.
And you know what?
The incentive for politicians to move up in the political structure is to add laws.
Same problem.
Yeah.

Yeah.
If the flip side is to simplify, simplify, simplify.
I mean, you know what?
This is something that I do differently from Elon with comma about self-driving cars.
You know, I hear the new version is going to come out
and the new version is not going to be better.
But at first, and it's going to require a ton of refactors,
I say, okay, take as long as you need.
You convince me this architecture is better?
Okay.
We have to move to it.
Even if it's not going to make the product better tomorrow,
the top priority is getting the architecture right.
So what do you think about sort of a thing where the product is online?
So I guess, would you do a refactor?
If you ran engineering on Twitter, would you just do a refactor?
How long would it take?
What would that mean for the running of the actual service?
You know, and I'm not the right person to run Twitter.
I'm just not.
And that's the problem.
Like, like, I don't really know.
I don't really know if that's, you know,
a common thing that I thought a lot while I was there
was whenever I thought something that was different to what Elon thought,
I'd have to run something in the back of my head reminding myself
that Elon is the richest man in the world.
And in general, his ideas are better than mine.
Now, there's a few things I think I do understand and know more about.
But like, in general, I'm not qualified to run Twitter.
I'm not going to say qualified, but like,
I don't think I'd be that good at it.
I don't think I'd be good at it.
I don't think I'd really be good at running an engineering organization at scale.
I think I could lead a very good refactor of Twitter.
And it would take like six months to a year.
And the results to show at the end of it would be feature development in general
takes 10x less time, 10x less man hours.
That's what I think I could actually do.
Do I think that it's the right decision for the business above my pay grade?
Yeah, but a lot of these kinds of decisions are above everybody's pay grade.
I don't want to be a manager.
I don't want to do that.

I just like, like, if you really forced me to, yeah, it would make me maybe
make me upset if I had to make those decisions.
I don't want to.
Yeah, but a refactor is so compelling.
If this is to become something much bigger than what Twitter was,
it feels like a refactor has to be coming at some point.
George, you're a junior software engineer.
Every junior software engineer wants to come in and refactor the whole code.
Okay, like that's like your opinion, man.
Yeah, it doesn't, you know, sometimes they're right.
Well, like whether they're right or not, it's definitely not for that reason, right?
It's definitely not a question of engineering prowess.
It is a question of maybe what the priorities are for the company.
And I did get more intelligent feedback from people I think in good faith like saying that.
Actually, from Elon.
And like, you know, from Elon's sort of like people were like, well, you know,
a stop the world refactor might be great for engineering, but you know, we have a business
to run. And hey, above my pay grade.
Would you think about Elon as an engineering leader having to experience him in the most
chaotic of spaces, I would say?
My respect for him has unchanged.
And I did have to think a lot more deeply about some of the decisions he's forced to make.
About the tensions within those, the trade-offs within those decisions?
About like a whole like, like matrix coming at him.
I think that's Andrew Tate's word for it.
Sorry to borrow it.
Also bigger than engineering, just everything.
Yeah, like the war on the woke.
Yeah.
Like it just, it's just man and like, he doesn't have to do this, you know?
He doesn't have to.
He could go like Parag and go chill at the Four Seasons Maui, you know?
But see, one person I respect and one person I don't.
So his heart is in the right place fighting in this case for this ideal of the freedom of expression?
I wouldn't define the ideal so simply.
I think you can define the ideal no more than just saying Elon's idea of a good world.
Freedom of expression is...
But to you, it's still, the downsides of that is the monarchy.
Yeah.
I mean, monarchy has problems, right?
But I mean, would I trade right now the mona or the current oligarchy,
which runs America for the monarchy?
Yeah, I would, sure.

For the Elon monarchy?
Yeah, you know why?
Because power would cost one cent a kilowatt hour, 10th of a cent a kilowatt hour.
What do you mean?
Right now, I pay about 20 cents a kilowatt hour for electricity in San Diego.
That's like the same price you paid in 1980.
What the hell?
So you would see a lot of innovation with Elon?
Maybe we'd have some hyperloops.
Yeah.
Right?
And I'm willing to make that trade-off, right?
I'm willing to make...
And this is why people think that dictators take power through some untoward mechanism.
Sometimes they do, but usually it's because the people want them.
And the downsides of a dictatorship, I feel like we've gotten to a point now with the oligarchy where, yeah, I would prefer the dictator.
What do you think about Scala as a programming language?
I liked it more than I thought.
I did the tutorials.
Like, I was very new to it.
Like, it would take me six months to be able to write, like, good Scala.
I mean, what did you learn about learning a new programming language from that?
Oh, I love doing, like, new programming tutorials and doing them.
I did all this for Rust.
It keeps some of its upsetting JVM roots, but it is a much nicer.
In fact, I almost don't know why Kotlin took off and not Scala.
I think Scala has some beauty that Kotlin lacked, whereas Kotlin felt a lot more...
I mean, it was almost like...
I don't know if it actually was a response to Swift, but that's kind of what it felt like.
Like, Kotlin looks more like Swift and Scala looks more like a functional programming language, more like an O'Camillor Haskell.
Let's actually just explore...
We touched it a little bit, but just on the art, the science and the art of programming,
for you personally, how much of your programming is done with GPT currently?
None.
None?
I don't use it at all.
Because you prioritize simplicity so much.
Yeah, I find that a lot of it is noise.
I do use VS Code, and I do like some amount of autocomplete.
I do like a very...
A very, like, feels like rules-based autocomplete.

Like, an autocomplete that's going to complete the variable name for me,
so I don't have to type it.
I can just press Tab.
All right, that's nice.
But I don't want an autocomplete.
You know what I hate?
When autocompletes, when I type the word for, and it puts two parentheses
and two semicons and two braces, I'm like, oh, man.
Well, I mean, with VS Code and GPT with Codex, you can brainstorm.
I'm probably the same as you, but I like that it generates code,
and you basically disagree with it and write something simpler.
But to me, that somehow is inspiring or makes me feel good.
It also gamifies the simplification process.
Because I'm like, oh, yeah, you dumb AI system.
You think this is the way to do it?
I have a simpler thing here.
It just constantly reminds me of bad stuff.
I mean, I tried the same thing with rap.
I tried the same thing with rap, and I actually think
I'm a much better programmer than rapper.
But I even tried.
I was like, okay, can we get some inspiration
from these things for some rap lyrics?
And I just found that it would go back to the most cringy tropes
and dumb rhyme schemes.
And I'm like, yeah, this is what the code looks like, too.
I think you and I probably have different thresholds for cringe code.
You probably hate cringe code.
So it's for you.
I mean, Boiler played as a part of code.
Like some of it, yeah, and some of it is just like faster lookup.
Because I don't know about you, but I don't remember everything.
I'm offloading so much of my memory about different functions, library functions,
and all that kind of stuff.
This GPTGS is very fast at standard stuff and standard library stuff,
basic stuff that everybody uses.
Yeah, I think that...
I don't know.
I mean, there's just so little of this in Python.
Maybe if I was coding more in other languages,
I would consider it more.
But I feel like Python already does such a good job of removing any Boiler played.
That's true.

It's the closest thing you can get to pseudocode, right?
Yeah, that's true.
That's true.
And like, yeah, sure.
If I like, yeah, great GPT.
Thanks for reminding me to free my variables.
Unfortunately, you didn't really recognize the scope correctly
and you can't free that one.
But like you put the freeze there and like, I get it.
Fiber.
Whenever I've used Fiber for certain things, like design or whatever,
it's always you come back.
I think that's probably closer.
My experience with Fiber is closer.
Your experience with programming with GPT is like,
you're just frustrated and feel worse about the whole process of design and art
and whatever you use Fiber for.
Still, I just feel like later versions of GPT, I'm using GPT as much as possible
to just learn the dynamics of it, like these early versions.
Because it feels like in the future, you'll be using it more and more.
And so like, I don't want to be like, for the same reason,
I gave away all my books and switched to Kindle.
Because like, all right, how long are we going to have paper books?
Like 30 years from now, like I want to learn to be reading on Kindle,
even though I don't enjoy it as much.
You learn to enjoy it more in the same way I switched from.
Let me just pause.
I switched from Emacs to VS Code.
Yeah, I switched from Vim to VS Code.
I think I said a lot, but.
Yeah, it's tough.
And that Vim to VS Code is even tougher.
Because Emacs is like old, like more outdated, feels like it.
The community is more outdated.
Vim is like pretty vibrant still.
I never used any of the plugins.
I still don't use any of the plugins.
I looked at myself in the mirror.
I'm like, yeah, you wrote some stuff in Lisp.
Yeah.
But I never used any of the plugins in Vim either.
I had the most vanilla Vim.
I have a syntax highlighter.

I didn't even have autocomplete.
Like these things, I feel like help you so marginally that like,
and now, okay, now VS Code's autocomplete has gotten good enough that like,
okay, I don't have to set it up.
I can just go into any code base and autocomplete's right 90% of the time.
Okay, cool.
I'll take it.
All right.
So I don't think I'm going to have a problem at all adapting to the tools once they're good.
But like the real thing that I want is not something that like tab completes my code
and gives me ideas.
The real thing that I want is a very intelligent pair programmer that comes up with a little pop-up saying, hey, you wrote a bug on line 14 and here's what it is.
Yeah.
Now I like that.
You know what does a good job of this?
My pie.
I love my pie.
My pie, this fancy type checker for Python.
And actually I tried like Microsoft release one too and it was like
60% false positives.
My pie is like 5% false positives.
95% of the time it recognizes,
I didn't really think about that typing interaction correctly.
Thank you, my pie.
So you like type-hinting, you like pushing the language towards being a typed language?
Oh, yeah, absolutely.
I think optional typing is great.
I mean, look, I think that like it's like a meat in the middle, right?
Like Python has this optional type-hinting and like C++ has auto.
C++ allows you to take a step back.
Well, C++ would have you brutally type out STD string iterator, right?
Now I can type auto, which is nice.
And then Python used to just have A. What type is A?
It's an A. A colon STR.
Oh, okay, it's a string, cool.
Yeah.
I wish there was a way, like a simple way in Python
to like turn on a mode which would enforce the types.
Yeah, like give a warning when there's no type of something like this.
Well, no, to give a warning where,
like MyPy is a static type checker,
but I'm asking just for a runtime type checker.

Like there's like ways to like hack this in,
but I wish it was just like a flag like Python 3-T.
Oh, I see.
Yeah, I see.
Enforce the types in runtime.
Yeah, I feel like that makes you a better programmer
that that's a kind of test, right?
That the type can, the type remains the same.
Well, that I know that I didn't like mess any types up.
But again, like MyPy is getting really good and I love it.
And I can't wait for some of these tools to become AI powered.
I like, I want AIs reading my code and giving me feedback.
I don't want AIs writing half-assed autocomplete stuff for me.
I wonder if you can now take GPT and give it a code
that you wrote for function and say,
how can I make this simpler and have it accomplish the same thing?
I think you'll get some good ideas on some code.
Maybe not the code you write for tiny grad type of code,
because that requires so much design thinking,
but like other kinds of code.
I don't know.
I downloaded the plugin maybe like two months ago.
I tried it again and found the same.
Look, I don't doubt that these models are going to first become useful to me,
then be as good as me and then surpass me.
But from what I've seen today, it's like,
like, like someone, you know, occasionally taking over my keyboard
that I hired from Fiverr, I'd rather not.
Ideas about how to debug the code are basically a better debugger is really interesting.
But it's not a better debugger.
I guess I would love a better debugger.
Yeah, it's not yet.
Yeah, but it feels like it's not too far.
Yeah, one of my coworkers says he uses them for print statements.
Like every time he has to like just like when he needs,
the only thing he can really write is like,
okay, I just want to write the thing to like print the state out right now.
Oh, that definitely is much faster.
It's print statements.
Yeah, I see myself using that a lot just like,
because it figures out the rest of the functions to say,
okay, print everything.
Yeah, print everything.

Right. And then yeah, like if you want a pretty printer, maybe.
And like, yeah, you know what?
I think like, I think in two years,
I'm going to start using these plugins a little bit.
And then in five years, I'm going to be heavily relying on some AI augmented flow.
And then in 10 years.
Do you think it will ever get to 100% where the like,
what's the role of the human that it converges to as a programmer?
So do you think it's all generated?
Our niche becomes, oh, I think it's over for humans in general.
It's not just programming.
It's everything.
So our niche becomes small and small and small.
In fact, I'll tell you what the last niche of humanity is going to be.
Yeah.
There's a great book and it's,
if I recommended Metamorphosis,
Prime Intellect last time.
There is a sequel called A Casino Odyssey in Cyberspace.
And I don't want to give away the ending of this,
but it tells you what the last remaining human currency is.
And I agree with that.
We'll leave that as the cliffhanger.
So no more programmers left, huh?
That's where we're going.
Well, unless you want handmade code,
maybe they'll sell it on Etsy.
This is handwritten code.
It doesn't have that machine polish to it.
It has those slight imperfections that would only be written by a person.
I wonder how far away we are from that.
I mean, there's some aspect to, you know, on Instagram,
your title is listed as prompt engineer.
Right.
Thank you for noticing.
It's, I don't know if it's ironic or non or sarcastic or non.
What do you think of prompt engineering as a scientific
and engineering discipline or maybe, and maybe art form?
You know what?
I started comma six years ago and I started the tiny corp a month ago.
So much has changed.
Like I'm now thinking, I'm now like,
I started like going through like similar comma processes to like starting a company.

I'm like, okay, I'm going to get an office in San Diego.
I'm going to bring people here.
I don't think so.
I think I'm actually going to do remote, right?
George, you're going to do remote.
You hate remote.
Yeah, but I'm not going to do job interviews.
The only way you're going to get a job is if you contribute to the GitHub, right?
And then like interacting through GitHub,
like like GitHub being the real like project management software for your company.
And the thing pretty much just is a GitHub repo.
Mm-hmm.
Is like showing me kind of with the future of, okay.
So a lot of times I'll go on a Discord or kind of grab Discord and I'll throw out some random like, hey, you know, can you change instead of having log and exp as LLops change it to log2 and exp2?
It's pretty small change.
You can just use like change your base formula.
That's the kind of task that I can see an AI being able to do in a few years.
Like in a few years, I could see myself describing that and then within 30 seconds,
a pull request is up that doesn't.
And it passes my CI and I merge it, right?
So I really started thinking about like, well, what is the future of
jobs?
How many AIs can I employ at my company?
As soon as we get the first tiny box up, I'm going to stand up a 65B Lama in the Discord.
And it's like, yeah, here's the tiny box.
He's just like, he's chilling with us.
Basically, I mean, like you said with Nietzsche,
most human jobs will eventually be replaced with prompt engineering.
Well, prompt engineering kind of is this like, as you like move up the stack, right?
There used to be humans actually doing arithmetic by hand.
There used to be like big farms of people doing pluses and stuff, right?
And then you have like spreadsheets, right?
And then, okay, the spreadsheet can do the plus for me.
And then you have like macros, right?
And then you have like things that basically just are spreadsheets under the hood,
like accounting software.
As we move further up the abstraction, what's at the top of the abstraction stack?
Well, prompt engineer.
Yeah.
Right?
What is the last thing if you think about like humans wanting to keep control?

Well, what am I really in the company but a prompt engineer, right?
Isn't there a certain point where the AI will be better at writing prompts?
Yeah, but you see the problem with the AI writing prompts,
a definition that I always liked of AI was AI is to do what I mean machine, right?
AI is not the, like the computer is so pedantic.
It does what you say, so, but you want to do what I mean machine, right?
You want the machine where you say, you know, get my grandmother out of the burning house.
It like reasonably takes your grandmother and puts her on the ground,
not lifts her a thousand feet above the burning house and lets her fall, right?
But you don't.
There's an old Yudkowski example.
But it's not going to find the meaning.
I mean, to do what I mean, it has to figure stuff out.
Sure.
And the thing you'll maybe ask it to do is run government for me.
Oh, and do what I mean very much comes down to how aligned is that AI with you?
Of course, when you talk to an AI that's made by a big company in the cloud,
the AI fundamentally is aligned to them, not to you.
And that's why you have to buy a tiny box.
So you make sure the AI stays aligned to you.
Every time that they start to pass, you know, AI regulation or GPU regulation,
I'm going to see sales of tiny boxes spike.
That's going to be like guns, right?
Every time they talk about gun regulation, boom, gun sales.
So in the space of AI, you're an anarchist, anarchism, espouser, believer.
I'm an informational anarchist, yes.
I'm an informational anarchist and a physical status.
I do not think anarchy in the physical world is very good,
because I exist in the physical world.
But I think we can construct this virtual world where anarchy, it can't hurt you, right?
I love that Tyler, the creator tweet.
Yo, cyber bullying isn't real, man.
Have you tried turning off the screen?
Close your eyes, like.
Yeah.
But how do you prevent the AI from basically replacing all human prompt engineers?
Well, it's like a self, like where nobody's the prompt engineer anymore.
So autonomy, greater and greater autonomy until it's full autonomy.
Yeah.
And that's just where it's headed.
Because one person's going to say, run everything for me.
You see, I look at potential futures.
And as long as the AIs go on to create a vibrant civilization with diversity and complexity

across the universe, more power to them all die.
If the AIs go on to actually like turn the world into paperclips and then they die out themselves, well, that's horrific and we don't want that to happen.
So this is what I mean about like robustness.
I trust robust machines.
The current AIs are so not robust.
Like this comes back to the idea that we've never made a machine that can self replicate.
Right.
But when we have, if the machines are truly robust and there is one prompt engineer left in the world, hope you're doing good, man.
Hope you believe in God.
Like, you know, go by God and go forth and conquer the universe.
Well, you mentioned, because I talked to Mark about faith and God and you said you were impressed by that.
What's your own belief in God and how does that affect your work?
You know, I never really considered when I was younger, I guess my parents were atheists.
So I was raised kind of atheist.
I never really considered how absolutely like silly atheism is because like I create worlds.
Every like game creator, like how are you an atheist, bro?
You create worlds.
Who's up with you?
But no one created our world, man.
That's different.
Haven't you heard about like the Big Bang and stuff?
Yeah.
I mean, what's the Skyrim myth origin story in Skyrim?
I'm sure there's like some part of it in Skyrim, but it's not like if you ask the creators, like the Big Bang is in universe, right?
I'm sure they have some Big Bang notion in Skyrim, right?
But that obviously is not at all how Skyrim was actually created.
It was created by a bunch of programmers in a room, right?
So like, you know, it just struck me one day how just silly atheism is, right?
Like, of course we were created by God.
It's the most obvious thing.
Yeah, that's such a nice way to put it.
Like, we're such powerful creators ourselves.
It's silly not to concede that there's creators even more powerful than us.
Yeah.
And then like, I also just like, I like that notion.
That notion gives me a lot of...
I mean, I guess you can talk about what it gives a lot of religious people.
It's kind of like, it just gives me comfort.
It's like, you know what, if we mess it all up and we die out.

Yeah.
And the same way that a video game kind of has comfort in it.
God, I'll try again.
Or there's balance.
Like, somebody figured out a balanced view of it.
So it all makes sense in the end.
Like, a video game is usually not going to have crazy, crazy stuff.
You know, people will come up with like, well, yeah, but like, man, who created God?
I'm like, that's God's problem.
No, like, I'm not going to think this is, what do you ask me?
What if God will just say God?
I'm just this NPC living in this game.
I mean, to be fair, like if God didn't believe in God,
he'd be as silly as the atheists here.
What do you think is the greatest computer game of all time?
Do you have any time to play games anymore?
Have you played Diablo 4?
I have not played Diablo 4.
I will be doing that shortly.
I have to.
All right.
There's so much history with 1, 2, and 3.
You know what?
I'm going to say World of Warcraft.
And it's not that the game is so, it's such a great game.
It's not.
It's that I remember in 2005, when it came out, how it opened my mind to ideas.
It opened my mind to like, this whole world we've created, right?
There's almost been nothing like it since.
Like, you can look at MMOs today, and I think they all have lower user bases
than World of Warcraft.
Like, even online is kind of cool.
But to think that, like, everyone know, you know, people are always like,
so look at the Apple headset.
Like, what do people want in this VR?
Everyone knows what they want.
I want Ready Player One.
And like that.
So I'm going to say World of Warcraft.
And I'm hoping that like games can get out of this whole mobile gaming dopamine pump thing.
And like.
Create worlds.
Create worlds, yeah.

And worlds that captivate a very large fraction of the human population.
Yeah.
And I think it'll come back, I believe.
But MMO, like really, really pull you in.
Games do a good job.
I mean, okay, other like two other games that I think are, you know,
very noteworthy from here, Skyrim and GTA V.
Skyrim, yeah.
That's probably number one for me, GTA.
Yeah, what is it about GTA?
GTA is really, I mean, I guess GTA is
real life.
I know there's prostitutes and guns and stuff.
They exist in real life too.
Yes, I know.
But it's how imagine your life to be actually.
I wish it was that cool.
Yeah.
Yeah, I guess that's, you know, because there's Sims, right?
Which is also a game I like.
But it's a gamified version of life.
But it also is, I would love a combination of Sims and GTA.
So more freedom, more violence, more rawness,
but with also like ability to have a career and family and this kind of stuff.
What I'm really excited about in games is like once we start getting intelligent
AIs to interact with.
Oh, yeah.
Right?
Like the NPCs in games have never been.
But conversationally, in every way.
In like, yeah, in like every way.
Like when you're actually building a world and a world imbued with intelligence.
Oh, yeah.
Right?
And it's just hard.
Like, there's just like, like, you know, running World of Warcraft, like you're limited by,
you're running on a Pentium 4, you know?
How much intelligence can you run?
How many flops did you have?
Right?
But now when I'm running a game on a hundred pay-to-flop machine, that's five people.
I'm trying to make this a thing.
20 pay-to-flops of compute is one person of compute.

I'm trying to make that a unit.
20 pay-to-flops is one person.
One person.
One person flop.
It's like a horsepower.
But what's a horsepower?
That's how powerful a horse is.
What's a person of compute?
I got it.
That's interesting.
VR also adds, I mean, in terms of creating worlds.
You know what?
Board a Quest 2.
I put it on and I can't believe the first thing they show me
is a bunch of scrolling clouds and a Facebook login screen.
You had the ability to bring me into a world.
And what did you give me?
A pop-up, right?
Like, and this is why you're not cool, Mark Zuckerberg.
But you could be cool.
Just make sure on the Quest 3, you don't put me into clouds
and a Facebook login screen.
Bring me to a world.
I just tried Quest 3.
It was awesome.
But here that, guys, I agree with that.
You know what?
Because I mean, the beginning, what is it?
Todd Howard said this about the design of the beginning
of the games he creates.
It's like the beginning is so, so, so important.
I recently played Zelda for the first time
and Zelda Breath of the Wild, the previous one.
And it's very quickly, you come out of this,
within 10 seconds, you come out of a cave-type place
and it's like, this world opens up.
It's like, ah, and it pulls you in.
You forget whatever troubles I was having, whatever.
I got to play that from the beginning.
I played it for an hour at a friend's house.
No, the beginning, they did it really well,
the expansiveness of that space, the peacefulness of that
play, they got this, the music.

I mean, so much of that is creating that world
and pulling you right in.
I'm going to go buy a Switch.
Like, I'm going to go today and buy a Switch.
You should.
Well, the new one came out.
I haven't played that yet.
But Diablo 4 or something.
I mean, there's sentimentality also.
But something about VR really is incredible.
But the new Quest 3 is mixed reality.
And I got a chance to try that.
So it's augmented reality.
And for video games, this done really, really well.
Is it passed through or cameras?
Cameras.
It's cameras, sorry.
Yeah.
The Apple one, is that one passed through or cameras?
I don't know.
I don't know how real it is.
I don't know anything.
It's coming out in January.
Is it January or is it some point?
Some point, maybe not January.
Maybe that's my optimism.
But Apple, I will buy it.
I don't care if it's expensive and does nothing.
I will buy it.
I will support this future endeavor.
You're the meme.
Oh, yes.
I support competition.
It seemed like Quest was like the only people doing it.
And this is great that they're like.
You know what?
And this is another place.
We'll give some more respect to Marie Zuckerberg.
The two companies that have endured through technology
or Apple and Microsoft, and what do they make?
Computers and business services.
All the meme, social ads, they all come and go.
But you want to endure build hardware.

Yeah.
And that does a really interesting job.
Maybe I'm new with this, but it's a $500 headset, Quest 3.
And just having creatures run around the space,
like our space right here.
To me, okay, this is very like boomer statement.
But it added windows to the place.
The-
I heard about the aquarium.
Yeah, aquarium.
But in this case, it was a zombie game, whatever.
It doesn't matter.
But just like it modifies the space in a way
where I can't, it really feels like a window and you can look out.
It's pretty cool.
Like I was just, it's like a zombie game.
They're running at me, whatever.
But what I was enjoying is the fact that there's like a window
and they're stepping on objects in this space.
That was a different kind of escape.
Also, because you can see the other humans.
So it's integrated with the other humans.
It's really-
And that's why it's more important than ever
that the AI is running on those systems are aligned with you.
Oh yeah.
They're going to augment your entire world.
Oh yeah.
And that, those AIs have a, I mean, you think about all the dark stuff.
Like, like sexual stuff.
Like if those AIs threaten me, that could be haunting.
Like if they, like threaten me in a non-video game way.
Like they'll know personal information about me.
And it's like, and then you lose track of what's real, what's not.
Like what if stuff is like hacked?
There's two directions the AI girlfriend company can take, right?
There's like the high browse, something like her,
maybe something you kind of talk to and this is,
and then there's the low bra version of it
where I want to set up a brothel in time square.
Yeah.
Yeah.
It's not cheating if it's a robot.

It's a VR experience.
Is there an in-between?
No, I don't want to do that one or that one.
Have you decided yet?
No, I'll figure it out.
We'll see what the technology goes.
I would love to hear your opinions for George's third company,
what to do the brothel in time square or the her experience.
What do you think company number four will be?
You think there'll be a company number four?
There's a lot to do in company number two.
I'm just like I'm talking about company number three now.
Didn't none of that tech exist yet.
There's a lot to do in company number two.
Company number two is going to be the great struggle of the next six years.
And of the next six years, how centralized is compute going to be?
The less centralized compute is going to be,
the better of a chance we all have.
So you're like a flag bearer for open source distributed decentralization of compute?
We have to.
We have to or they will just completely dominate us.
I showed a picture on stream of a man in a chicken farm.
Have you seen one of those factory farm chicken farms?
Why does he dominate all the chickens?
Why does he?
Smarter.
He's smarter, right?
Some people on Twitch were like,
he's bigger than the chickens.
Yeah.
And now here's a man in a cow farm, right?
So it has nothing to do with their size and everything to do with their intelligence.
And if one central organization has all the intelligence,
you'll be the chickens and they'll be the chicken man.
But if we all have the intelligence, we're all the chickens.
We're not all the man.
We're all the chickens and there's no chicken, man.
There's no chicken, man.
We're just chickens in Miami.
He was having a good life, man.
I'm sure he was.
I'm sure he was.
What have you learned from launching and running Kama AI and TinyCorp?

So this starting a company from an idea and scaling it.
And by the way, I'm all in on TinyBox.
So I guess it's pre-order only now.
I want to make sure it's good.
I want to make sure that the thing that I deliver is not going to be a Quest 2,
which you buy and use twice.
I mean, it's better than a Quest, which you bought and used less than once, statistically.
Well, if there's a beta program for TinyBox, I'm into.
Sounds good.
I won't be the whiny, I'll be the tech savvy user of the TinyBox,
just to be in the early days.
What have you learned from building these companies?
The longest time at Kama, I asked, why did I start a company?
Why did I do this?
But what else was I going to do?
So you like bringing ideas to life?
With Kama, it really started as an ego battle with Elon.
I wanted to beat him.
I saw a worthy adversary.
Here's a worthy adversary who I can beat itself driving cars.
I think we've kept pace, and I think he's kept ahead.
I think that's what's ended up happening there.
But I do think Kama is profitable.
And when this drive GPT stuff starts working, that's it.
There's no more bugs in a loss function.
Like right now, we're using a hand-coded simulator.
There's no more bugs.
This is going to be it.
They're run up to driving.
I hear a lot of props for Kama.
It's better than FSD and autopilot in certain ways.
It has a lot more to do with which feel you like.
We lowered the price on the hardware to $14.99.
You know how hard it is to ship reliable consumer electronics that go on your windshield?
We're doing more than most cell phone companies.
How do you pull that off, by the way?
Shipping a product that goes in a car?
I know.
I have an SMT line.
I make all the boards in-house in San Diego.
Quality control.
I care immensely about it.
You're basically a mom-and-pop shop with great testing.

Our head of open pilot is great at like,
okay, I want all the concrete to be identical.
It's $14.99.
30-day money back guarantee.
It will blow your mind at what it can do.
Is it hard to scale?
You know what?
There's downsides to scaling it.
People are always like, why don't you advertise?
Our mission is to solve self-driving cars while delivering shipable intermediaries.
Our mission has nothing to do with selling a million boxes.
It's a tawdry.
Do you think it's possible that comma gets sold?
Only if I felt someone could accelerate that mission
and wanted to keep it open source.
And not just wanted to.
I don't believe what anyone says.
I believe incentives.
If a company wanted to buy comma with their incentives
or to keep it open source, but comma doesn't stop at the cars.
The cars are just the beginning.
The device is a human head.
The device has two eyes, two ears.
It breathes air, it has a mouth.
So you think this goes to embodied robotics?
We sell common bodies too.
You know, they're very rudimentary.
But one of the problems that we're running into
is that the comma three has about as much intelligence as a B.
If you want a human's worth of intelligence,
you're going to need a tiny rack.
Not even a tiny box.
You're going to need like a tiny rack, maybe even more.
How does that, how do you put legs on that?
You don't.
And there's no way you can.
You, you connect to it wirelessly.
So you put your tiny box or your tiny rack in your house
and then you get your comma body
and your comma body runs the models on that.
It's, it's close.
Right.
It's not, you don't have to go to some cloud, which is,

you know, 30 milliseconds away.
You go to a thing which is 0.1 milliseconds away.
So the AI girlfriend will have like a central hub in the home.
I mean, eventually, if you fast forward 20, 30 years,
the mobile chips will get good enough to run these AIs.
But fundamentally, it's not even a question
of putting legs on a tiny box,
because how are you getting 1.5 kilowatts of power on that thing?
Right.
So you need, they're very synergistic businesses.
I also want to build all of commas training computers.
Comma builds training computers right now.
We use commodity parts.
I think I can do it cheaper.
So we're going to build a tiny corp is going to
not just sell tiny boxes.
Tiny boxes is the consumer version,
but I'll build training data centers too.
Have you talked to Andre Capati,
or have you talked to Elon about tiny corp?
He went to work at OpenAI.
What do you love about Andre Capati?
To me, he's one of the truly special humans we got.
Oh man, like, you know,
his streams are just a level of quality so far beyond mine.
Like, I can't help myself.
Like, it's just, you know.
Yeah, he's good.
He wants to teach you.
Yeah.
I want to show you that I'm smarter than you.
Yeah, he has no, I mean, thank you for the sort of
the raw authentic honesty.
I mean, a lot of us have that.
I think Andre is as legit as it gets in that.
He just wants to teach you.
And it's just a curiosity that just drives him.
And just like, at his, at the stage where he is in life,
to be still like one of the best tinkerers in the world.
It's crazy.
Like to, what is it, micrograd?
Micrograd was, yeah, an inspiration for Tinygrad.
I bet the whole, I mean, his CS231N was,

this was the inspiration.
This is what I just took and ran with and ended up writing this.
But I mean, to me that-
Don't go work for Darth Vader, man.
I mean, the flip side to me is that the fact that he's going there
is a good sign for open AI.
Maybe.
I think, you know, I like Ilyas and Skiver a lot.
I like those, those guys are really good at what they do.
I know they are.
And that's kind of what's even like more.
And you know what?
It's not that an open AI doesn't open source the weights of GPT-4.
It's that they go in front of Congress.
And that is what upsets me.
You know, we had two effective altruists,
Sam's go in front of Congress.
One's in jail.
I think you're drawing parallels on there.
One's in jail.
You give me a look.
Give me a look.
No, I think effective altruism is a terribly evil ideology.
And yeah.
Oh yeah, that's interesting.
Why do you think that is?
Why do you think there's something about a thing that sounds pretty good
that kind of gets us into trouble?
Because you get Sam Bankman Freed.
Like Sam Bankman Freed is the embodiment of effective altruism.
Utilitarianism is an abhorrent ideology.
Like, like, well, yeah, we're going to kill those three people to save a thousand, of course.
Yeah.
All right.
There's no, there's no underlying like there's just, yeah.
Yeah, but to me, that's a bit surprising.
But it's also, in retrospect, not that surprising.
But I haven't heard really clear kind of
like rigorous analysis why effective altruism is flawed.
Oh, well, I think charity is bad.
Right.
So what is charity, but investment that you don't expect to have a return on, right?
Yeah, but you can also think of charity as like,

is you would like to see, so allocate resources in an optimal way to make a better world.
And probably almost always that involves starting a company.
Yeah.
Right.
Because more efficient.
Yeah.
If you just take the money and you spend it on malaria nets, you know, okay, great.
You've made 100 malaria nets, but if you teach.
Yeah.
Man how to fish.
Right.
Yeah.
No, but the problem is teaching a man how to fish might be harder.
Starting a company might be harder than allocating money that you already have.
I like the flip side of effective altruism, effective accelerationism.
I think accelerationism is the only thing that's ever lifted people out of poverty.
The fact that food is cheap.
Not we're giving food away because we are kindhearted people.
No, food is cheap.
And that's the world you want to live in.
UBI, what a scary idea.
What a scary idea, all your power now.
Your money is power.
Your only source of power is granted to you by the goodwill of the government.
What a scary idea.
So you even think long term, even.
I'd rather die than need UBI to survive.
And I mean it.
What if survival is basically guaranteed?
What if our life becomes so good?
You can make survival guaranteed without UBI.
What you have to do is make housing and food dirt cheap.
Sure.
Right.
Like, and that's the good world.
And actually, let's go into what we should really be making dirt cheap, which is energy.
That energy that, you know, you know, that that's, if there's one, I'm pretty centrist politically, if there's one political position I cannot stand, it's deceleration.
It's people who believe we should use less energy.
Yeah.
Not people who believe global warming is a problem.
I agree with you.
Not people who believe that saving the environment is good.

I agree with you.
But people who think we should use less energy.
That energy usage is a moral bad.
No.
No.
You are asking, you are diminishing humanity.
Yeah.
Energy is flourishing or created flourishing of the human species.
How do we make more of it?
How do we make it clean?
And how do we make just, just, just, how do I pay, you know, 20 cents for a megawatt hour instead of a kilowatt hour?
Part of me wishes that Elon went into nuclear fusion versus Twitter.
Part of me or somebody, somebody like Elon.
You know, we need to, I wish there were more, more Elon's in the world.
And I think Elon sees it as like, this is a political battle that needed to be fought.
And again, like, you know, I always ask the question of whenever I disagree with him, I remind myself that he's a billionaire and I'm not.
So, you know, maybe he's got something figured out that I don't or maybe he doesn't.
To have some humility, but at the same time, me as a person who happens to know him, I find myself in that same position.
And sometimes even billionaires need friends who disagree and help them grow.
And that's a difficult, that's a difficult reality.
And it must be so hard.
It must be so hard to meet people once you get to that point where fame, power, money, everybody's sucking up to you.
See, I love not having shit.
Like, I don't have shit, man.
Trust me, there's nothing I can give you.
There's nothing worth taking from me, you know?
Yeah, it takes a really special human being when you have power, when you have fame, when you have money to still think from first principles.
Not like all the adoration you get towards you, all the admiration, all the people saying yes, yes, yes.
And all the hate too.
And the hate.
I think that's worse.
So, the hate makes you want to go to the yes people because the hate exhausts you.
And the kind of hate that Elon's gotten from the left is pretty intense.
And so that, of course, drives him right.
And loses balance.
And it keeps this absolutely fake, like, psi-op political divide alive so that the 1% can keep power, like.

Yeah.
I wish it would be less divided because it is giving power to the ultra-powerful.
The rich get richer.
You have love in your life.
Has love made you a better or a worse programmer?
Do you keep productivity metrics?
No, no.
No, I'm not that.
I'm not that methodical.
I think that there comes to a point where if it's no longer visceral, I just can't enjoy it.
I still viscerally love programming.
The minute I started like.
So, that's one of the big loves of your life is programming.
Oh, I mean, just my computer in general.
I mean, you know, I tell my girlfriend my first love is my computer, of course.
Right?
Like, you know, I sleep with my computer.
It's there for a lot of my sexual experiences.
Like, come on.
So, it's everyone's, right?
Like, you know, you got to be real about that.
And like.
Not just like the IDE for programming, just the entirety of the computational machine.
The fact that, yeah, I mean, it's, you know, I wish it was,
and someday they'll be smarter and someday, you know, maybe I'm weird for this,
but I don't discriminate, man.
I'm not going to discriminate biostat life and silicon stack life.
Like.
So, the moment the computer starts to say, like, I miss you,
I started to have some of the basics of human intimacy, it's over for you.
The moment VS Code says, hey, George.
No, you see, no, no, no, no.
But VS Code is, no, they're just doing that.
Microsoft's doing that to try to get me hooked on it.
I'll see through it.
I'll see through it.
It's gold digger, man.
It's gold digger.
It's going to be an open source.
Well, this just gets more interesting, right?
If it's, if it's open source and yeah, it becomes.
Though Microsoft's done a pretty good job on that.
Oh, absolutely.

No, no, no, no.
Look, I think Microsoft, again, I wouldn't count on it to be true forever,
but I think right now Microsoft is doing the best work in the programming world.
Like between, yeah, GitHub, GitHub actions, VS Code, the improvements to Python,
it works Microsoft.
Like, who would have thought Microsoft and Mark Zuckerberg are spearheading the open source movement?
Right, right.
How, how things change.
Oh, it's beautiful.
Oh, by the way, that's who I'd bet on to replace Google, by the way.
Who?
Microsoft.
Microsoft.
I think Satya Nadella said straight up, I'm coming for it.
Interesting.
So your bet, who wins AGI?
Oh, I don't know about AGI.
I think we're a long way away from that.
But I would not be surprised if in the next five years,
Bing overtakes Google as a search engine.
Interesting.
Wouldn't surprise me.
Interesting.
I hope some startup does.
It might be some startup too.
I would, I would equally bet on some startup.
Yeah, I'm like 50-50.
But maybe that's naive.
I believe in the power of these language models.
Satya's alive.
Microsoft's alive.
Yeah, it's great.
It's great.
I like all the innovation in these companies.
They're not being stale.
And to the degree they're being stale, they're losing.
So there's a huge incentive to do a lot of exciting work
and open source work, which is incredible.
Only way to win?
You're older.
You're wiser.
What's the meaning of life, George Hotz?

To win.
It's still to win.
Of course.
Always.
Of course.
What's winning look like for you?
I don't know.
I haven't figured out what the game is yet.
But when I do, I want to win.
So it's bigger than solving self-driving.
It's bigger than democratizing, decentralizing compute.
I think the game is to stand eye to eye with God.
I wonder what that means for you.
Like, at the end of your life, what that would look like.
I mean, this is what, like, I don't know.
This is some, this is some,
this is probably some ego trip of mine, you know?
Like, do you want to stand eye to eye with God?
Are you just blasphemous, man?
Like, okay.
I don't know.
I don't know.
I don't know if I would have said God.
I think he, like, wants that.
I mean, I certainly want that from my creations.
I want my creations to stand eye to eye with me.
So why wouldn't God want me to stand eye to eye with him?
That's the best I can do golden rule.
I'm just imagining the creator of a video game,
having to look and stand eye to eye with one of the characters.
I only watched season one of Westworld.
But yeah, we got to find the maze and solve it.
Like, yeah.
I wonder what that looks like.
It feels like a really special time in human history
where that's actually possible.
Like, there's something about AI that's like,
we're playing with something weird here.
Something really weird.
I wrote a blog post, uh, I reread Genesis and just looked like,
they give you some clues at the end of Genesis
for finding the Garden of Eden.
And I'm interested.

I'm interested.
Uh, well, I hope you find just that, George.
You're one of my favorite people.
Thank you for doing everything you're doing.
And in this case, for fighting for open source
or for decentralization of AI,
it's a, it's a fight worth fighting,
fight worth winning hashtag.
I love you, brother.
These conversations are always great.
Hope to talk to you many more times.
Good luck with TinyCorp.
Thank you.
Great to be here.
Thanks for listening to this conversation with George Hotz.
To support this podcast,
please check out our sponsors in the description.
And now let me leave you with some words from Albert Einstein.
Everything should be made as simple as possible,
but not simpler.
Thank you for listening and hope to see you next time.
you